CA 1® Tape Management

Utilities and Reports Reference Guide

Release 12.6.00

technologies

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.

CA Technologies Product References

This document references the following CA Technologies products:

- CA 1® Tape Management (CA 1)
- CA 7® Job Management (CA 7)
- CA ACF2™ for z/OS (CA ACF2)
- CA ASM2® Backup and Restore(CA ASM2)
- CA Disk® Backup and Restore (CA Disk)
- CA Earl™ (CA Earl)
- CA Roscoe® Interactive Environment (CA Roscoe)
- CA TLMS® Tape Management (CA TLMS)
- CA Top Secret® Security for z/OS (CA Top Secret)
- CA Vantage™ Storage Resource Manager GMI (CA Vantage GMI)

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: Introduction 15

Chapter 2: Utilities 33

Appendix A: Report Cross-Reference 531

Index 543

Chapter 1: Introduction

This section contains the following topics:

About this Guide

This guide provides information on all CA 1 utilities. A functional description of each program, JCL and parameter information, control statement specifications, and sample reports are supplied.

Note: For information on the messages generated the CA 1 Tape Management utilities, see the *Messages Reference Guide*.

Summary of Utilities

This following CA 1 utilities are available:

CTSDEU

Used to erase residual data from a tape's prior use.

CTSPMTST

Used to test the character strings or patterns that you have created. Each is checked for conformity and proper syntax. A report is issued to aid in resolving invalid pattern errors.

CTSSYNC

Allows the synchronization between CA 1 and the 3495/3494 Library Control System (LCS) databases.

CTSTAPER

Tests CA 1 operating system intercepts and other functions.

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.

DYNFORMT

Formats the common communication queues used by the CA Dynam/T interface.

DYNRELSE

Releases the CA Dynam/T interface's common communications queue that has remained 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 and generates input for updating the databases.

DYNUPDTE

Processes the output from DYNSYNC used to update the TMC to resolve discrepancies for the CA Dynam/T interface.

L0C6INIT

Identifies a subcomponent provided to interface CA 1 installation and initialization functions with the CA Common Services CAIRIM component.

TMSAGGR

Maintains DSNB record chains in the TMC.

TMSAUDEX

This sub program of the TMSXTEND reads the audit file to determine the systems sharing the TMC and their maintenance level. It can also be run in standalone mode to provide this information in the summary format.

TMSAUDIT

Provides a way to analyze and report on TMC activity. This program allows you to specify aspects of the report with control statements.

TMSBINQ

Produces reports about the format of the TMC and Audit, or the content of records in the TMC. This program allows you to specify aspects of the reports.

TMSBLDUE

Builds TMMVOLDF statements for use with the CA 1 supplied exits TMSXITU and TMSXITE.

TMSBLDVR

This subprogram of TMSXTEND builds a new TMC when adding or removing volume ranges or DSNBs. It can also be executed to create and initialize a new TMC.

TMSCKDTM

Verifies that packed decimal fields, such as date and time, are valid in the TMC.

TMSCKLVL

Reports CA 1 maintenance levels of key processing modules. Useful when contacting CA 1 support.

TMSCLEAN

Processes and expires tapes under Days Since Last Used Control, resets the expiration date for Catalog Days Control to Catalog Control, resets Catalog Control to a real expiration date, and produces reports listing new scratch volumes and scratch volume inventory.

TMSCLNOA

Processes output from TMSCLEAN with clean criteria and out-of-area criteria.

TMSCONVR

Reads SMF and other user input data sets to create transaction files for processing by TMSUPDTE, TMSFORMT, TMSGRW and TMSEXPDT.

TMSCOPY

Backs up the TMC/Audit and restores the TMC.

TMSCTLG

Processes and removes tapes from Catalog Control.

TMSCYCLE

Processes and removes tapes from Cycle Control.

TMSDATA

Creates sequential data sets from TMC volume and DSNB records for use in TMSGRW processing or for programs written in-house.

TMSDELET

Sets Volume records to active or inactive status and lists volumes in inactive status.

TMSDISP

Performs Batch Auxiliary Disposition processing.

TMSEXPDT

Applies retention to data sets from the Retention Data Set (RDS).

TMSFORMT

Formats/extends the TMC or formats the Audit file.

TMSFVSN

Initializes all first volume serial number fields (DSNBFVSN) in all DSNB records which is required for security processing. This utility need not be run for users at Release 5.1 or later as the DSNBFVSN is always updated at these levels.

TMSGRW

Selects TMC or Audit records to print reports in a customized format.

TMSIDATA

Processes lists of volume information or MVS catalog listings to create transactions to be used to create a new TMC when no tape management system is available to convert.

TMSINIT

Activates, batch activates, deactivates or reactivates CA 1 and initiates CA 1 health checks.

TMSINITD

Validates CA 1 hlq.CTAPOPTN Scratch Pool Management members.

TMSLBLPR

Creates external tape label requests using the Audit file.

TMSMERGE

Merges data provided from TMSSPLIT to another TMC.

TMSMULV

Constructs multivolume pointers for TMC Volume records.

TMSOSCAT

Compares the system catalog to the TMC and identifies the differences. It can also be used to set the OSC bit in each volume or DSNB. This may need to be done after a conversion from another tape management product to CA 1.

TMSPTRS

Checks the multivolume, multi-data set and DSNB free chains to verify the internal integrity of the TMC.

TMSPULL

Produces library pull and setup lists for tape volume prestaging.

TMSREMOV

Physically removes Volume records from the TMC.

TMSRINIT

TMC Extend monitor.

TMSSMF

Captures tape activity data from an SMF extract file and formats it as TMSAGGR and TMSUPDTE transactions.

TMSSMSUP

Obtains SMS Management Class and updates the TMC.

TMSSPLIT

Selects data from a TMC for use by TMSMERGE.

TMSSTATS

Reports the CA 1 options in *hlq*.CTAPOPTN presently active in the operating system.

TMSTPNIT

Performs tape volume header label initialization.

TMSTPPRO

Verifies tape VOLSER header labels, constructs TMC records, and erases sensitive data from scratched tapes.

TMSUDSNB

Provides batch updates to DSNB/Control Records.

TMSUNCAT

Uses output file from TMSCLEAN to uncatalog data sets on other systems.

TMSUPDTE

Provides batch updates to Volume records.

TMSVMEDT

Validates the Vault Pattern Description (VPD) data set for errors and selects TMC records to be processed by the Vault Management System.

TMSVMUPD

Updates the TMC to reflect tape volume movement for the Vault Management System.

TMSVMVLT

Processes the validated and reformatted Vault Pattern Description (VPD) data set and the TMC records passed from TMSVMEDT to determine the Vault Management System's movement of volumes.

TMSXTEND

The Dynamic TMC Extend utility, allows you to add or remove volume ranges or DSNB records, reblock or move the TMC to a new volume while tape processing is active.

Some of the above programs produce reports that can be minimally controlled via PARMs on the EXEC statement. Most reports are written using CA Earl and can be easily customized.

Access and Responsibilities

We recommend that some of the utilities described in this guide should be executed only by the systems programmer responsible for maintaining CA 1. Instructions to this effect are included in the description of the utility when appropriate. We also recommend restricted access to certain programs with powerful or time-consuming functions, such as TMSFORMT.

Prerequisites

The initialization of CA 1 into an active or batch active state with the TMSINIT utility is a prerequisite for most CA 1 utilities. The discussion of each utility notes which states are acceptable for program execution.

All CA 1 utilities which update the TMC will abend with U0008 if the audit threshold, which is specified in the CA 1 system options, is met. Ensure that there are enough free audit records prior to running these CA 1 utilities.

Override the Preferred Date Pattern

The DATEFMT parameter provided in hlq.CTAPOPTN gives you the ability to change the system default date pattern yyyy/ddd to your own *preferred date pattern*. This pattern determines how external dates are entered, printed and displayed. The preferred date pattern specified in *hlq*.CTAPOPTN can be overridden by a unique date pattern during execution of most CA 1 utilities by using the following parameter in the JCL for that utility:

```
PARM='DATEFMT=(fmt)'
```

DATEFMT=(fmt) is an optional parameter which overrides the DATEFMT value specified in hlq.CTAPOPTN if CA 1 is active or batch active, or the Julian default date pattern yyyy/ddd if CA 1 has never been initialized.

The date pattern entered for (fmt) specifies how dates appear in reports. The value of (fmt) is a new date pattern. If DATEFMT=(fmt) is not specified, the default date pattern specified in *hlq*.CTAPOPTN is taken.

Important! Several programs (TMSEXPDT, TMSGRW, TMSUDSNB, TMSUPDTE, and TMSVMEDT) allow you to specify actual dates within control statements. If a non-Julian date format is used within control statements and you change your preferred date format, you must change the control statement dates accordingly.

We recommend that actual dates within control statements be maintained in Julian date format. This allows the job to execute properly even if you change your preferred date format.

Date Format Pattern

The pattern indicated for (fmt) is limited to 10-bytes and must be enclosed in parentheses when entered on an EXEC statement of a CA 1 program. It must consist of all of the elements from any one group (1-3) listed below, and can contain any combination of elements from the separator character group. Groups 1, 2 and 3 contain the elements of a complete date. They can be arranged in any order, but each element must be used only *once*.

The following date format groups can be used:

- Group 1: YYYY DDD
- Group 2: YYYY MM DD
- Group 3: YYYY MMMDD

DDD

Specifies a Julian day.

Range: 000 to 366

DD

Specifies a day of the month. The range of days in January is 00-31. The range of days in February is 01-28, except during leap year when the range is 01-29. The range of days in December is 01-32, except during leap year when the range is 01-31. January 0 and December 32 are acceptable dates since the JCL allows values such as EXPDT=98000 and EXPDT=97366.

Range: 0 to 32

MM

Specifies a number for the month.

Range: 01 to 12

YYYY

Specifies a valid year. Dates can be specified in the range January 0, 1960 (1960/000), through the 21st Century date, December 32, 2155 (2155/366).

Range: 1960 to 2155

MMM

Specifies the month.

Valid values: JAN, FEB, MAR, APR, MA,Y JUN, JUL, AUG, SEP, OCT, NOV, and DEC.

Separators

The following table shows how separators are used to format dates.

Separators	Examples	Results
period (.)	DDMMM.YYYY	01JUN.2005
slash (/)	YYYY/DDD	2005/152
dash (-)	MM-DD-YYYY	06-01-2005
comma (,)	MMMDD,YYYY	JUN01,2005
space ( )	DD MM YYYY	01 06 2005

The last entry under *Results* (01 06 2005) illustrates the importance of using the proper format to enter dates. The date format tells CA 1 whether to process 01 06 2005 as January 6, 2005 or June 1, 2005.

Delimiters

To update TMC or DSNB EXPDT values, dates that can be entered as PARMs or in control statements can be accepted with or without delimiters. A delimiter is a special character such as a single quote. The delimiter appears in the first position of a value and at the end of the value. For example:

```
EXPDT=DDMMMYYYY
EXPDT='DDMMMYYYY'
```

Important! Expiration dates can be specified with or without delimiters. However, if delimiters are not used, it is recommended that slashes, periods, or dashes be used as separators instead of commas, because commas are used as separators between parameter specifications. Dates that contain commas or blanks must be specified within delimiters to allow the program to correctly interpret the syntax of the statement.

The DATEFMT Command

The CA 1 online facilities, CA 1 ISPF and TIQ Online Inquiry/Update, can also be used to specify DATEFMT. If you are using CA 1 ISPF (the recommended facility for inquiring on and updating TMC records), you can enter your preferred date format on the CA 1 ISPF Session Options Screen. This date format is stored in your profile pool so that it can be used across sessions.

You can use the DATEFMT command available in TIQ to specify your preferred date format.

Note: For more information about these facilities, see the *Administration Guide*.

The value you specify for DATEFMT has no effect on the heading line at the top of a report. All report headings, regardless of the DATEFMT specified, resemble the following example:

```
SUNDAY, March 25, 2005.176        YOUR  COMPANY  NAME     PAGE=00001
```

The date format which appears in most of the report samples in this guide was defined by specifying the following parameter to the programs:

```
PARM='DATEFMT=(MMMDD YYYY)'
```

This date format also appears in examples:

```
U,109472,EXPDT='APR15 2001'
```

The date format, MMMDD YYYY, was selected for illustration purposes because it is easy to read and interpret.

Keyword Expiration and Retention

CA 1 provides expiration and retention for tape data sets through specific criteria or unique Julian date keywords specified in the JCL LABEL parameter, JCL ACCODE parameter, or in a Retention Data Set control statement.

Note: For information on the format of the RDS expiration and retention specification, see Control Statement Specification (see page 247).

Since many problems can occur when using the EXPDT JCL parameter including conflicts between dates and CA 1 options and restrictions by various IBM facilities, you can specify some options through the ACCODE JCL parameter. Although ACCODE is primarily for ANSI labeled tapes, it is available for all processing. To allow compatibility with ANSI labeled tape, the first character can be any letter. If the next two characters are CA, the remainder is used to select the expiration or retention option. If both an ACCODE and an EXPDT/RETPD value were specified, the ACCODE takes precedence when assigning an expiration date in the TMC and the EXPDT/RETPD value takes precedence updating the HDR1 record on the physical tape.

The following specifications can be used in the JCL LABEL parameter:

LABEL=RETPD=0

A temporary data set. If it is for the first file, the volume remains in scratch status after use. If the *hlq*.CTAPOPTN parameter WRKFLS is set to NO, all temporary data sets are assigned a one day retention period.

LABEL=EXPDT=yyddd

Standard system Julian expiration date. (This format is not valid for the Year 2000 and beyond.)

LABEL=EXPDT=yyyy/ddd

Standard MVS expiration date for DFP 2.3 and above.

LABEL=RETPD=dddd

Standard system retention period.

LABEL=EXPDT=88uuu or ACCODE=xCAUSER

User Control EXPDT keyword that you can create. Specifies that a tape is held permanently.

Note: x must be replaced with any letter.

LABEL=EXPDT=90ddd

Catalog Days Control EXPDT keyword. Specifies that a tape is held at least ddd days. Once ddd has been met, the expiration date is changed to Catalog Control and is protected as long as the data set is cataloged.

Range: 001 to 365

LABEL=EXPDT=98000 or ACCODE=xCANORES

Nonresident EXPDT keyword. Specifies that the tape volume being processed is not under CA 1 control.

Note: x must be replaced with any letter.

LABEL=EXPDT=98ddd

Days Since Last Used Control EXPDT keyword. Specifies that a tape data set is held as long as it is used every ddd days. As long as a tape data set is being utilized at least as frequently as every ddd days, it is protected. When the data set is not used for ddd days, it becomes eligible for scratch. This is an excellent means of controlling tapes used for testing.

Range: 001 to 366

LABEL=EXPDT=99000 or ACCODE=xCACATLG

Catalog Control EXPDT keyword. Specifies that a tape data set is protected as long as it has an entry in the system catalog. As soon as CA 1 determines that the data set no longer has an entry in the system catalog, it becomes eligible for scratch. 99000 is frequently used for generation data groups.

Note: x must be replaced with any letter.

LABEL=EXPDT=99365 or LABEL=EXPDT=99366 or ACCODE=xCAPERM

Never Scratch EXPDT keyword. Specifies that a tape is held permanently. Data sets held with this keyword cannot be recreated on the same tape; however, these data sets can have additional information MODed on. 99365 is often considered to be manual control, as these tapes can not be scratched until the volume is manually expired.

Note: x must be replaced with any letter.

LABEL=EXPDT=99ccc

Cycle Control EXPDT keyword. Specifies that ccc cycles of a data set are to be held. This retention is based entirely on the number of cycles to be maintained in the TMC. Cycle Control can be used for simple data sets and generation data groups. For GDGs, the number of tapes retained by CA 1 does not necessarily need to equal the number of MVS catalog entries. The definition of a CA 1 cycle is determined by CA 1 system options. Cycle control only uses the first file on the tape in the determination of the number of cycles that have been created.

Note: If an expiration date supplied in the JCL is less than or equal to the system date, and is not a keyword date, then CA 1 arbitrarily sets the TMC EXPDT to STATS/001. This is treated as permanent retention and a manual update would need to be made to reset the retention to what is actually desired. If a temporary data set is specified but a retention or expiration date into the future is specified, the tape is not considered temporary by CA 1.

Specify CA 1 Keywords

Listed below are the JCL Julian format of CA 1 EXPDT keywords and the corresponding CA 1 batch and online keyword expression of these values:

blank

A date with an internal value of zeros is always displayed as the word ZEROS. However, the date can only be entered as any number of 0s, blanks, nulls, or the word ZEROS.

Equivalent keyword: ZEROS

yyyy/ddd

Julian date format. CA 1 validates all input dates to ensure that they fall within this range.

Range: 1960/001 to 2155/366

Equivalent keyword: yyyy/ddd

Note: For more information, see Considerations (see page 28).

88*uuu*

This keyword allows you to create your own keywords for processing. CA 1 treats this as PERMANENT.

Equivalent keyword: USER/*uuu*

ACCODE: xCAUSER

90*ddd*

Retain *ddd* days, then retain while data set is cataloged to the operating system.

Equivalent keyword: CATLG/*ddd*

98000

This is a nonresident tape, so do not update the TMC. This value is not stored in the TMC but appears in the Audit data set for type 3 (exception) records. CA 1 does not allow you to set the expiration of a data set to FOREIGN in a TMC record.

Equivalent keyword: FOREIGN

ACCODE: xCANORES

98*ddd*

Retain *ddd* days after date on which tape was last used.

Range: 0001 to 366

Equivalent keyword: LDATE/ddd

99000

Retain while data set is cataloged to the operating system.

Equivalent keyword: CATLG or CATALOG

ACCODE: xCACATLG

99*ccc*

Retain ccc cycles.

Range: 001 through 365.

Equivalent keyword: CYCLE/*ccc*

99365,99366

Retain data set permanently.

Equivalent keywords: PERM or PERMANENT

ACCODE: xCAPERM

STATS/sss

Status of held tape where sss is the reason code indicating why the tape is being held. This keyword has no JCL equivalent. It is set by programs through the SET_KEYWORD function, or is entered through the keyword format STATS/sss, to apply permanent retention other than 99365 to an unknown situation, such as a broken chain.

Equivalent keyword: STATS/001

The EXPDT in the JCL is less than the current date; therefore, CA 1 set the EXPDT to STATS/001 to prevent the tape from scratching prematurely.

Equivalent keyword: STATS/002

There is a multivolume chaining error. A TMC volume with this EXPDT is chained to a volume not in the TMC.

Equivalent keyword: STATS/003

The EXPDT in the TMC for the volume is updated as STATS/003 when ACCODE=xCANORES is specified in the JCL.

Note: An x in the first position of the ACCODE must be replaced by any letter.

Considerations

EXPDT=yyddd date supplied in the JCL falls within the CA 1 EXPDT keyword range, it is treated as a keyword.

When ACCODE is specified with a valid combination for CA 1, EXPDT is not interpreted. If ACCODE=xCAEXPDT, the expiration date is treated as an expiration date regardless of any CA 1 definition. If ACCODE=xCAKEYWD, the expiration date is treated as a keyword.

TMS Banner Pages

TMS banner pages similar to the sample shown below are automatically generated for some CA 1 reports using the TMSRPT ddname.

```
           TTTTTTTTTTTT
           TTTTTTTTTTTT
           TTTTTTTTTTTT                  .....             .....                   .SSSSSSS.
           TTTTTTTTTTTTTTTTTTTTTT'   .MMMMMMMMMMM.     .MMMMMMMMMMM.            .SSSSSSSSSSSSS.
           TTTTTTTTTTTTTTTTTTTT'   .MMMMMMMMMMMMMMMM.MMMMMMMMMMMMMMMM.        .SSSSSSSSSSSSSSSSS.
           TTTTTTTTTTTTTTTTTTT'  .MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM.     .SSSSSSSSSSSSSSSSSSS.
           TTTTTTTTTTTTTTTTTT'  .MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM.    SSSSSSSSSSSSSSSSSSSSS.
           TTTTTTTTTTTTTTTTTT   MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM   .SSSSSSSSSSSSSSSSSSSSSS
           TTTTTTTTTTTTTTTTTT  .MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM.  SSSSSSSSSS'   'SSSSSSSS.
           TTTTTTTTTTTT        MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM  'SSSSSSSSS.      'SSSSS'
           TTTTTTTTTTTT        MMMMMMMMMMMM' 'MMMMMMMMMMMMM' 'MMMMMMMMMMMM   SSSSSSSSSSS.       'SS
           TTTTTTTTTTTT        MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM   'SSSSSSSSSSSS.
           TTTTTTTTTTTT        MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM    SSSSSSSSSSSSSSS.
           TTTTTTTTTTTT        MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM     'SSSSSSSSSSSSSSSS.
           TTTTTTTTTTTT        MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM       'SSSSSSSSSSSSSSSS
           TTTTTTTTTTTT        MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM          'SSSSSSSSSSSSSS
           TTTTTTTTTTTT        MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM   SS.       'SSSSSSSSSSS.
           TTTTTTTTTTTT.       MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM  .SSSSS.      'SSSSSSSSSS
           TTTTTTTTTTTTTTTTTT  MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM  'SSSSSSSS.   .SSSSSSSSSS
           TTTTTTTTTTTTTTTTTT  MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM   SSSSSSSSSSSSSSSSSSSSSS'
            TTTTTTTTTTTTTTTTT  MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM   'SSSSSSSSSSSSSSSSSSSSS
            'TTTTTTTTTTTTTTTT  MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM    'SSSSSSSSSSSSSSSSSS'
              'TTTTTTTTTTTTTT  MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM     'SSSSSSSSSSSSSSSS'
                'TTTTTTTTTTTT  MMMMMMMMMMMM   MMMMMMMMMMMMM   MMMMMMMMMMMM       'SSSSSSSSSSSS'
                                                                                    'SSSSSS'
                          C A - 1   -   T A P E   M A N A G E M E N T   S O F T W A R E
                                       V E R S I O N   5   L E V E L   2
                                         Y O U R   C O M P A N Y   N A M E
                                      Y O U R   C O M P A N Y   A D D R E S S
                                         Y O U R   C O M P A N Y   C I T Y

CA 1 IS A PROPRIETARY SOFTWARE PACKAGE LICENSED FROM CA, INC. FOR THE SOLE USE OF THIS
   INSTALLATION.   ALL OTHER RIGHTS ARE RESERVED.   USE OF THIS SOFTWARE BY UNAUTHORIZED PERSONS IS PROHIBITED.

                TMS REPORT=nn  JOB=jobname   STEP=stepname       TUESDAY, March 07, 2005.128           TIME=16:44a39
```

To suppress TMS banner pages for CA 1 reports, use the procedure described on Suppressing TMS Banner Pages (see page 30).You can also use this procedure to suppress the TMS FLAG Bit Definition Page (shown on TMS FLAG Bit Definition Page (see page 31)) that appears in certain reports.

The number of lines per page and the record length of the CA 1 report (TMSRPT DD) may also be changed.

Suppressing TMS Banner Pages

To alter the logo (banner pages), line count, record length or reporting of FLAG definitions in the job to be executed:

1. Determine the report number to be altered. Report numbers are shown in the heading line of the report and are listed in the appendix "Report Cross-Reference (see page 531)."
2. Insert a TMSOPTNS DD in the JCL that produces the report. It should refer to an 80-byte record length data set containing the following data:

   1

   REPORT (or * for comment statements)

   7-8

   Report number (Reports 1 through 9 should be specified without a leading zero. A space should be in position 8.)

   10

   Option=Value[,Option=Value][, Option=Value][,Option=Value]

   LOGO YES or NO

   LINECNT 1 through 99

   LRECL Decimal value

   FLAGS YES - print the TMS FLAG Bit Definition Page (applies to certain reports); NO - do not print the TMS FLAG Bit Definition Page

Example: Banner Page Suppression

```
REPORT49 LOGO=NO,LINECNT=60,LRECL=80,FLAGS=NO
```

A sequential data set (or PDS member) can be set up to contain values for all reports, if desired. Then each report job can have a TMSOPTNS DD pointing to a common *control* set. One such member is provided in the *hlq*.CTAPOPTN library as member name TMSOPTNS.

TMS FLAG Bit Definition Page

```
TUESDAY, MAY 07, 2005.128   Y O U R   C O M P A N Y   N A M E

TMS REPORT-32   AUDIT RECORDS


                             DEFINITION OF TMC FLAG-1 BITS

                                80 = INTERNAL FIELD CHANGED BY USER
                                40 = VOLUME CLOSED NORMALLY
                                20 = TMC RECORD UPDATED BY USER
                                10 = VOLUME CLOSED BY ABEND
                                08 = VOLUME LISTED TO BE CLEANED
                                04 = VOLUME LISTED IN SCRATCH STATUS
                                02 = VOLUME IN DELETE (INACTIVE) STATUS
                                01 = VOLUME ELIGIBLE FOR RDS OVERRIDE

                             DEFINITION OF TMC FLAG-2 BITS

                                80 = DATA SET WAS ON MVS CATALOG WHEN
                                READ
                                40 = VOLUME OPENED FOR OUTPUT
                                20 = DATA SET RECREATED
                                10 = TEMPORARY DATA SET
                                08 = EXPIRED FROM CATALOG CONTROL
                                (TMSCTLG)
                                04 = EXPIRED FROM CYCLE CONTROL
                                (TMSCYCLE)
                                02 = EXPIRED FROM LDATE CONTROL
                                (TMSCLEAN)
                                01 = EXPIRED BY TMS, CA 11 OR EDM

                             DEFINITION OF TMC FLAG-3 BITS

                                80 = CA-9/R+ INDICATED BAD TAPE
                                40 = TAPE RELEASED BY EXT VAULT MANAGER
                                20 = CONTROLLED BY EXTERNAL DATA MANAGER
                                10 = CONTROLLED BY CA DYNAM/T
                                08 = DATA SET ERASE REQUIRED
                                04 = DEFAULT EXPDT USED AT OPEN
                                02 = MULTIPLE FILES EXIST
                                01 = CREATED BY CA 1/COPYCAT

                             DEFINITION OF TMC FLAG-4 BITS

                                80 = TAPE HAS BEEN EXPIRED BY SMS
                                40 = ACTUAL VOLSER IN USE
                                20 = VAULT SPECIFIC REQUEST
                                10 = TAPE HAS BEEN DEGAUSSED
                                08 = FILE ON OS CATALOG
                                04 = NON-RESIDENT TAPE
                                02 = TAPE IS IN USE FOR REALTIME
                                STACKING
                                01 = NO FURTHER STACKING IS ALLOWED ON
                                TAPE

                             DEFINITION OF TMC FLAG-5 BITS

                                80 = TAPE IS USED FOR REALTIME STACKING
```

Dynamic Allocation of CA 1 TMC and Audit Data Sets

The TMC and Audit data sets are dynamically allocated for every batch program that requires the same TMC or Audit that is being used by realtime processing. A check is included to ensure that the TMC or Audit data sets have not been moved or recataloged since CA 1 became active. For these programs, a DD statement of TMC or AUDIT is not allowed.

For those programs that do not require an active TMC, it is dynamically allocated if a TMC DD statement is not part of the JCL. However, if a TMC DD statement is present in the JCL, that statement is used. For a few programs (TMSFORMT, for example) a TMC DD statement is still required.

Chapter 2: Utilities

This chapter provides a functional description of each program, JCL and parameter information, control statement specifications, and sample reports.

CTSDEU Utility—Erase Data

Use the CTSDEU Data Erase batch utility to erase residual data from a tape's prior use following one or more valid files.

The CTSDEU utility writes pattern data to the end of the tape which prevents any previous data from being recovered and ensures that no confidential data is accidentally transmitted. The new tape media types include servo tracks that are used for error correction and integrity purposes and on some models may also include metadata about the files on the volume. These new media types cannot be erased using the bulk degauss process therefore, the CTSDEU provides the ability to erase only the data on the volume without destroying the servo tracks.

The CTSDEU utility is only invoked in batch. Use this utility to:

- Execute the batch JCL in a standalone job, referencing the volume to be erased or
- Include a step in a production job stream to routinely perform a data erase on a tape that is created in a preceding step.

When a multivolume data set is being secured, normally only the last volume needs to be secured because the other volumes in the set are full of data. The ALL parameter is provided to request that all volumes in the multivolume set be mounted and a security erase performed. This is normally only necessary if the program that created the tape issues a Force End of Volume command to stop writing on a volume and to cause a new volume to be mounted.

This utility identifies virtual tape volumes and rejects the request to perform an ERASE action on them. If you are trying to perform a full volume ERASE on a 3590-H the ERASE function may take more than 40 minutes, a 3592-JA may take as much as 1.5 hours. If the wait-time timeout value is less than this for the job, a S522 abend will occur.

Job Control Statements

This feature is delivered as a single program CTSDEU which is installed in the CA 1 load library CAI.CTAPLINK when maintenance is applied using SMP/E. Sample JCL to execute the CTSDEU program follows:

```
//stepname  EXEC  PGM=CTSDEU[,PARM='parm']
//STEPLIB   DD   DSN=CAI.CTAPLINK,DISP=SHR
//SYSPRINT  DD SYSOUT=*
//SYSUT1    DD   UNIT=(TAPE,,DEFER),DISP=(OLD,KEEP),LABEL=(,SL),  *NOTES
//               VOL=SER=anyvol,                                  *NOTES
//               DSN=any.tape.file
```

Notes:

- LABEL=(,BLP) can be used when trying to erase a scratch. Use LABEL=(,SL) on all other processing of CTSDEU.
- The JCL parameter VOL=SER= is required if the requested DSN is not in the MVS catalog.
- If you are trying to perform a FULL VOLUME ERASE on a 3590-H type device, then the ERASE function can possibly take more than 40 minutes. If the wait-time timeout value is less for this job, a S522 abend can occur.

Parameter Definition

ERASE

Reads the tape header label and verifies that the tape is in the TMC, that it is not a virtual tape, and the data set names match those recorded in the TMC. If the volume serial number was not supplied by VOLLAB1, the VOLSER from the VOL1 record on SL or AL or TMSXTPPR user exit. The console operator is asked to supply the volume serial number.

Note: When ERASE is used on a scratch tape, we assume that the intent is to erase the whole tape. The DSN field in the tape catalog will be changed to (CTSDEU.ERASE.yyyy/ddd), where the first character is a hexzero and the tape will be reinitialized.

NOERASE

Reads the tape header label and verifies that the tape matches the TMC, no residual data is erased. The default is ERASE.

Note: If the NOERASE parameter is used on a scratch tape, the specified scratch tape is taken out of scratch status. Only the volume specified in the JCL will be mounted.

ALL

Indicates that all volumes of the requested tape file should be secured. Normally when a set of tapes are produced, they are filled completely, overlaying any old data. If force end of volume (FEOV) macro is used to produce short tapes, data previously written may not be overlaid.

LAST

Indicates that only the last volume of the requested tape file will be secured. The default is ALL.

LIST

A detail list of the contents of the requested volumes is produced.

NOLIST

A detail list is not produced. The default is NOLIST.

CTSMAIL Utility – Email or WTO

The CTSMAIL utility enables you to send WTOs and emails that contain substitution variables from the VDB. The utility is the batch version of the services that the Volume Pool Monitor (VPM) subtask of the CTS provides. CTSMAIL can do the following actions:

- Create a WTO or email from a source in the CAMAILIN DD statement.
- Create one or more WTOs or emails from a member list in the EXEC PARM.
- Create one or more WTOs or emails from a member list in the CAMAILIN DD statement.

CTSMAIL enables you to test the formatting of emails or WTOs that you create. You can then specify the emails and WTOs in an alert that the VPM subtask monitors. You can also add notification emails or WTOs to existing jobs for tape management and other systems.

JCL to execute the CTSMAIL utility is provided in member CTSJMAIL of *hlq*.CTAPJCL.

Note: Product variables that the TMSCLEAN, TMSCOPY, and TMSVMUPD utilities create are available to include in emails or WTOs after the VPM subtask updates the Product Variables table and writes it back to the VDB. The OPT_VPM_TIMER variable specifies how often the VPM subtask updates the table. If OPT_VPM_TIMER specifies 10 minutes, it can be up to 10 minutes before the new product variables are added to the VDB and information from the most recent run of these utilities is available.

CTSMAIL invokes either a WTO processor (CTSWTO) or an email processor (CTSEML) depending on the member that the CAI.SEND data set or the CAMAILIN DD processes. CTSMAIL obtains a list of substitution variables and their values and it changes the WTO or email text to reflect the variable values. The utility then issues the WTO or sends the email.

The source statements are in a data set that the CAMAILIN DD defines or in a CAI.SEND data set that the variable OPT_EMAIL_DEFAULT_DSN defines. If the source is in the CAI.SEND data set, the utility obtains a member list from the EXEC PARM or CAMAILIN DD statement. If you specify both the items, the utility uses only the EXEC PARM.

If you specify a member list, each CAI.SEND member is dynamically allocated. CTSMAIL then processes the source statements for control statements. The remaining statements are tabled for the WTO or email processor. If the source is in-stream in the CAMAILIN DD, CAI.SEND is not allocated.

Note: The output of the WTO is limited to 426 lines of 70 characters by z/OS restrictions. The tabled source is limited to 60,000 bytes. Email servers between the sender and receiver can limit the output of the email.

Report Description

CTSMAIL does not produce a report. CTSMAIL and its processors (CTSWTO, CTSEML) write WTOs and set return codes when errors occur.

Job Control Statements

```
//SEND     EXEC PGM=CTSMAIL,
//  PARM='SAMPE01,SAMPW01'
//STEPLIB  DD DSN=CAI.CTAPLINK,DISP=SHR
//SYSUDUMP DD   SYSOUT=*
//*
//*CAMAILIN DD   DSN=CAI.SEND(SAMPE01),DISP=SHR  SOURCE IN A PDS MEMBER
//*CAMAILIN DD   DSN=CAI.SEND(SAMPW01),DISP=SHR  SOURCE IN A PDS MEMBER
//*CAMAILIN DD   DSN=ANY.FILE,DISP=SHR           SOURCE IN A FILE
//*CAMAILIN DD   *                               SOURCE INLINE
//CAMAILIN DD   *
  SAMPW01,SAMPE01,YOURMAIL,YOURWTO       /* MANY MEMBER OF ONE CARD */
  SAMPW01                                /* ONE MEMBER ON EACH CARD */
  SAMPE01                                /*        OR               */
  YOURMAIL                               /*   A COMBINATION         */
  SAMPW01 SAMPE01,YOURMAIL    YOURWTO    /* SEPARATED WITH COMMAS    */
  SAMPW01                     SAMPE01    /*     OR SPACES           */
//
```

Parameter Definition

member,member,...

Specifies a list of members. The members must exist in the CAI.SEND data set.

JCL Considerations

CAMAILIN DD

Do not specify if you use the EXEC PARM to specify a member of the CAI.SEND data set to process. If you do not use EXEC PARM, specify the CAMAILIN DD to provide a list of members in the CAI.SEND data set to process. The CAMAILIN can also include source statements to create a WTO or an email.

Control Statement Specification

You can specify the control statements in the CAMAILIN DD in two formats. You can specify them as a list of members to process in the CAI.SEND data set. If the first statement begins with EMAIL or WTO, you define an in-stream email or WTO to send.

For emails, the first statement contains just EMAIL followed by email header fields and the body of the email.

For WTOs, the first statements must contain WTO with the ROUTCDE parameter on the same line or the next line.

The following example shows a test WTO with an in-stream text:

```
//CAMAILIN DD *
WTO
ROUTCDE=(1,3,5,11)
TEXT
This is a test WTO sent from &SYSID& on &SYSDATE& to show the value of the
&TMSCOPY_DATE_OF_LAST_BACKUP& variable.
/*
```

The following example shows a test WTO with a list of members. All the members must be in the CAI.SEND data set:

```
//CAMAILIN DD *
EMAIL001, MAIL2RB
MAIL0050
WTO045
/*
```

Note: For more information about constructing an email, see the *Administration Guide*.

CTSSYNC Utility—Support 3495/3494 Devices

Use the CTSSYNC utility to support 3495/3494 devices.

The CTSSYNC command provides a high level interface to the IBM 3495/3494 LCS (Library Control System) and optionally lists the results of client requested functions. The CTSSYNC utility performs the following functions to support 3495/3494 devices:

- Synchronizes the tape management catalog system and the 3495/3494 database. The status of a volume is obtained from the tape management database. The status is set in the 3495/3494 LCS database.
- Ejects cartridges needed for off-site storage from the 3495/3494. Tapes are ejected to the BULK or Convenience locations.
- Changes the status of cartridges. The status is set in the 3495/3494 LCS database according to the specified command without reference to the tape management database.
- Allows IMPORT/EXPORT of virtual tapes from the 3495/3494. Virtual tapes are extracted from or added to physical container tapes that can be moved in or out of the 3495/3494 robot.

This utility accepts input either via a PARM value and/or SYSIN control statements. The PARM value can be either from a calling program or the EXEC JCL statement. Commands specified without VOLSERs are "global" commands. These global commands are used for subsequent VOLSERs that are specified without commands. Global commands are replaced by subsequent global commands but not a specific command that includes a VOLSER.

The default global command is SYNC. Each command is processed when the VOLSER or VOLSER range is specified independent of any previous or subsequent commands.

Job Control Statements

```
//stepname      EXEC  PGM=CTSSYNC,PARM='parm'
//STEPLIB       DD    DISP=SHR,DSN=CAI.CTAPLINK
//SYSPRINT      DD    SYSOUT=*                                    *NOTES
//ERREPORT      DD    SYSOUT=*                                    *NOTES
//EXPORT        DD    DSN=EXPORT.TAPE.LIST,                       *NOTES
                      UNIT=VTAPE,TRTCH=NOCOMP,DISP=(,CATLG)
//IMPORT        DD    DSN=IMPORT.TAPE.LIST,                       *NOTES
                      UNIT=VTAPE,TRTCH=NOCOMP,DISP=(,CATLG)
//SYSIN         DD    *                                           *NOTES
(control statements here)
/*
```

Notes:

- The SYSPRINT DD is optional. The CTSSYNC command writes the standard or full report to this file if coded. The standard report produces a detailed list of the requested commands for each volume and the results of their requests. When the command is for a range of volumes, a line is listed for each volume. For SYNC commands the results of the tape management query is also shown.
- The ERREPORT DD is optional. The optional ERREPORT DD Statement produces an Errors Only report in the same format as the Standard report.
- The EXPORT DD statement is required by the EXPORT command. CTSSYNC writes all the files required by the 3494/VTS for an EXPORT Operation to the virtual tape defined by this DD statement. If no EXPORT Commands are specified, the DD is not used. Do not specify 'VOL=(,RETAIN)' on this DD statement.

  Do not specify 'VOL=(,RETAIN)' on this DD statement. Specifying VOL=(,RETAIN) causes the EXPORT list volume to remain on the tape drive. The robot does not consider the list volume available if it is mounted, so the EXPORT invoked by CTSSYNC fails.
- The IMPORT DD statement is required by the IMPORT command. CTSSYNC writes all the files required by the 3494/VTS for an IMPORT Operation to the virtual tape defined by this DD statement. If no import Commands are specified DD is not used. Do not specify 'VOL=(,RETAIN)' on this DD statement.

  Do not specify 'VOL=(,RETAIN)' on this DD statement. Specifying VOL=(,RETAIN) causes the IMPORT list volume to remain on the tape drive. The robot does not consider the list volume available if it is mounted, so the IMPORT invoked by CTSSYNC fails.
- The SYSIN DD is optional. You can enter commands by using the SYSIN control statements or by coding them as parameters after the PARM= statement, or by passing a parameter list with the use of Register 1. Commands are free form, but the entire command line must be on a single line between columns 1 and 71. Spaces and commas are delimiters. Comments must have the form /* comment text */. They may appear anywhere within a command or on a line by themselves.
- To run the CTSSYNC command, CA 1 must be active or batch active.
- The CTSSYNC utility cannot be used with 3495/3494 BTLS devices.

Control Statements

The CTSSYNC utility processes the following commands:

- VERIFY
- NOVERIFY
- BULK
- NOBULK

- FORCE
- NOFORCE
- EJECT[,vvvvvv[-vvvvvv]]
- PROTECT[,vvvvvv[-vvvvvv]]
- SCRATCH[,vvvvvv[-vvvvvv]]
- PRIVATE[,vvvvvv[-vvvvvv]]
- SYNC[,vvvvvv[-vvvvvv]]
- EXPORT[,llllll[-llllll]] [,destination]
- IMPORT[,pppppp] [,llllll[-llllll]] [,ooooooo], where vvvvvv is volume serial number (real or virtual), llllll is volume serial number for a logical (virtual) tape, pppppp is volume serial number for a physical (container) tape and ooooooo is IMPORT option either SCRATCH or PRIVATE.
- SYNCTEST[,vvvvvv[-vvvvvv]]
- COPY,POOL[,pp] where pp is the physical volume pool that contains the logical volumes to export.

VERIFY|NOVERIFY

The VERIFY command instructs the system to verify whether the volume serial number is defined in the 3495/3494 LCS database before executing any further commands. If the NOVERIFY command is specified, the LCS commands are issued even if the cartridge does not reside in the 3495/3494 LCS database. The VERIFY command is the default.

BULK|NOBULK

The BULK command instructs any of the following EJECT commands to be sent to the bulk location. The NOBULK command instructs all of the following EJECT commands to be sent to the convenience location. If neither BULK nor NOBULK commands are specified prior to an EJECT command, the utility defaults to the bulk location.

FORCE|NOFORCE

The FORCE command instructs the program to execute any following SYNC commands unconditionally, regardless of the current use attribute in the TCDB. The NOFORCE command causes SYNC commands to be processed only if the use attribute in the TCDB does not match the volume status in the tape database, or if TCDB or Library Manager show an error indicator. NOFORCE is the default.

EJECT

The EJECT command instructs the 3495/3494 robotic devices to eject the specified VOLSER(s). Depending on whether a BULK or NOBULK commands is specified prior to the EJECT command, the cartridge is ejected to either the bulk or the convenience location.

PROTECT|SCRATCH|PRIVATE

The PROTECT, SCRATCH, and PRIVATE commands all change the status of the specified VOLSER(s). The SCRATCH command changes the status to scratch, while the PROTECT and PRIVATE commands change the status to private within the 3495/3494 LCS database. This affects both cartridge(s) that are selected to satisfy SCRATCH requests, and cartridge(s) that are added to auto cartridge loaders (ACLs) located inside the 3495/3494 robotic devices. There is no difference between PROTECT and PRIVATE commands.

SYNC

The SYNC command causes the specified VOLSER(s) to be synchronized between the tape management database and the 3495/3494 LCS database. If the status is different, the tape management database is assumed to be correct. This is the default if no specific command is specified.

EXPORT

The EXPORT command causes the specified LOGICAL VOLSER(s) to be written to the export list volume, a logical volume residing in the same library as the volumes to be exported. After all volumes have been written to the export list volume, the OAM LCS schedules the list to be exported.

DESTINATION

This sub-parameter can only be included on the EXPORT control statement and is optional. If included, all virtual volumes with the same destination are stacked together on a single physical volume. If a list of 50 virtual volumes to be exported is supplied; and 25 has one destination and the other 25 has a different destination, then the first 25 would be stacked together on one set of physical volume(s) and the other 25 would be stacked on a different set of physical volume(s). However, this value is not sent to the CBRUXEJC exit during eject processing nor is it tracked in the TCDB or the tape management catalog.

IMPORT

The IMPORT command causes the specified LOGICAL VOLSER(s) to be written to the import list volume, a logical volume residing in the same library as the volumes to be imported. After all volumes have been written to the import list volume, the OAM LCS schedules the list to be imported.

Note: Only 1 EXPORT or IMPORT command can be scheduled at a time. Subsequent EXPORTs or IMPORTs are shown as errors by CTSSYNC utility.

SCRATCH|PRIVATE

One of these two optional sub-parameters can be included on the IMPORT command control statement. If the specific virtual volume(s) are specified, the default PRIVATE indicates that the specified virtual volume should be copied back into cache inside the 3494/VTS completely. The entire virtual volume is copied). If SCRATCH command is specified, then the specified virtual volume is assumed to be in scratch status and only the VOL1/HDR1/HDR2 records are copied back into cache. This process reduces the time needed to complete the IMPORT process. The virtual volumes cannot be made un-scratched, since the data on them no longer resides inside the 3494/VTS.

ABDLMT=

ABDLMT command is a keyword parameter that is used to set a return code limit at which CTSSYNC will ABEND. After CTSSYNC utility has completed all other processing and is about to terminate, the numeric value of ABDLMT is compared with the highest return code from any command. If that return code is greater than or equal to the ABDLMT, CTSSYNC utility abends with U0006 and a reason code equal to the return code. The default is ABDLMT=99. It can be in either the ABDLMT=nn or the ABDLMT(nn) form. It can be specified in the EXEC PARM field or in a SYSIN statement. It can be specified on a statement alone or with other commands. It can be specified multiple times, but only the last value is used.

SYNCTEST

Performs like the SYNC command but does not call to update the OAM. Both the TCDB (VOLCAT) and the Library Manager (LM) database are compared with the TMC too. If the status of the volume is not synchronized in all three data bases or catalogs, an error is reported to both the SYSPRINT and ERREPORT DD statements.

COPY

Performs like the EXPORT command except that the exported volumes are copied, not moved. List the VOLSERs to be exported below the COPY command in the SYSIN DD.

Note: Specify the POOL parameter before you export the volumes.

POOL

Specifies the physical volume pool that contains the logical volumes to export. This command is only used with the COPY command.

Examples of Specifying Commands to CTSSYNC

There are three methods of getting the commands or VOLSERs to the CTSSYNC utility: PARM= on the EXEC statement, through SYSIN control statements, or passed in a parameter list through Register 1. The easiest way to perform a single command is through use of the PARM statement on the EXEC command.

```
//STEP1 EXEC PGM=CTSSYNC,PARM='EJECT,123456'
```

This causes cartridge 123456 to be ejected from the 3495/3494 robotic device. The same value could have been used as input from the SYSIN control statements.

```
//SYSIN  DD *
EJECT,123456
```

The command can also be passed in a parameter list pointed by Register 1. This is used if another program, such as the security exit wants to call the synchronization program passing it the command. The format of the parameter list pointed by Register 1 is as follows:

```
R1  ==>  A(addr1)  ==>  CL8'EJECT'
         A(addr2)  ==>  CL8'123456'
```

It is possible to use the SYSIN control statement in addition to either of the two-parameter methods. For example;

```
//STEP1 EXEC PGM=CTSSYNC,PARM='BULK'
//SYSIN  DD *
EJECT,123456
EJECT,100001
EJECT,001234
```

In this example, the BULK command applies to all subsequent commands. Therefore, the EJECT of tapes 123456, 100001, and 001234 are directed to the bulk location. If a volume command is issued without a VOLSER, all subsequent VOLSERs are then processed.

```
//STEP1  EXEC PGM=CTSSYNC
//SYSIN  DD *
SYNC
100002
102030
100202
```

In this example, the three listed volumes (100002, 102030, and 100202) have their status synchronized between the tape management and 3495/3494 LCS database. This could just as easily be a list of tapes to be ejected to the BULK location, as a post process to the vaulting system for example.

```
//STEP1    EXEC PGM=CTSSYNC
//EXPORT   DD   DSN=EXPORT.TAPE.LIST,
//                UNIT=VTAPE,TRTCH=NOCOMP,DISP=(,CATLG)
//SYSIN    DD *
EXPORT,LV1234,OFFSITE1           /* comments */
EXPORT,LV2345,OFFSITE1
```

In this example, the logical volumes LV1234 and LV2345 are to be removed. An export list volume is created and the 3495/3494 LCS schedules the list to be exported. When the export function completes, logical volumes LV1234 and LV2345 are stacked on to a physical volume. The physical volume is put in export hold status, waiting to be removed from the 3495/3494.

```
//STEP1    EXEC PGM=CTSSYNC
//IMPORT   DD   DSN=IMPORT.TAPE.LIST,
//              UNIT=VTAPE,TRTCH=NOCOMP,DISP=(,CATLG)
//SYSIN    DD *
IMPORT,PV0001,LV1234
IMPORT,PV0001,LV2345-LV2348       /* range of logical volumes */
```

In this example, logical volumes LV1234 and LV2345 through LV2348 are imported from physical volume PV0001. An import list volume is created and the 3495/3494 LCS schedules the list to be imported.

```
//STEP1  EXEC PGM=CTSSYNC
//IMPORT DD   DSN=IMPORT.TAPE.LIST,
//            UNIT=VTAPE,TRTCH=NOCOMP,DISP=(,CATLG)
//SYSIN  DD *
IMPORT,PV0001
```

In this example, all logical volumes on physical volume PV0001 are to be imported. An import list volume is created and the 3495/3494 LCS schedules the list to be imported.

Note: The CTE3495 member in CTAPECPB can be used to generate a list of VOLSERs as input to CTSSYNC.

```
//STEP1  EXEC PGM=CTSSYNC,PARM='SCRATCH,ABDLMT=12'
//IMPORT DD   DSN=IMPORT.TAPE.LIST,
//            UNIT=VTAPE,TRTCH=NOCOMP,DISP=(,CATLG)
//SYSIN  DD *
012345        /* SINGLE TAPE WITH COMMENT */
ABC017-ABC052  /* SCRATCH ALL TAPES IN THIS RANGE */
SYNC XYZ100-XYZ299
```

In this example, the EXEC PARM is used to set 'SCRATCH' as a global command and to ABEND CTSSYNC if the highest return code is greater than or equal to 12. The VOLSER 012345 and the range of VOLSERs ABC017 through ABC052 will be scratched using the global command. The tape management system will get synchronized with VOLSERs ranging from XYZ100 to XYZ299.

The FORCE command tells CTSSYNC to synchronize the VOLSER 123456 regardless of the current use attribute in the TCDB. The following examples show how to use the FORCE control statement:

Examples

```
//STEP1 EXEC PGM=CTSSYNC,PARM='FORCE'
//SYSIN DD  *
SYNC,123456
```

or

```
//STEP1 EXEC PGM=CTSSYNC,PARM='SYNC'
//SYSIN DD  *
FORCE,123456
```

These examples show how to use the FORCE control statement. The FORCE command tells CTSSYNC to sync the volser 123456 regardless of the current use attribuie in the TCDB.

Completion Codes:

CTSSYNC can complete with the following condition codes.

RC=0

Successful completion: The requested commands performed successfully or commands were not processed because no VOLSER was provided.

RC=4

Successful completion with redundant commands: In VERIFY mode, one or more ATL commands failed or could not be performed because the VOLSER was not defined in the 3495/3494 LCS database or for the SYNC command, synchronization of one or more volumes was bypassed, or a PRIVATE, PROTECT or SCRATCH command was issued for a volume, which was already in the requested status for LCS database.

RC=8

Processing errors: One or more ATL commands failed or could not be performed because the VOLSER was not defined in the 3495/3494 LCS database, and NOVERIFY was specified, or no valid input command was found, or the tape management system was not active.

RC=12

Incorrect job setup: A required DD statement is missing.

RC=16

Invalid request: A schedule of both IMPORT and EXPORT operations was attempted.

User Abends:

A U0006 abend is issued when CTSSYNC finishes processing all control statements and is going to terminate with a condition code greater than the value specified in the abend limit PARM (ABDLMT=). For example: ABDLMT=8 will cause CTSSYNC to abend with a U0006-rc when the highest RC is greater than or equal to 8.

When the 3495/3494 LCS is unable to schedule an EXPORT or IMPORT, CTSSYNC abends with the following user abend codes:

U0100-00rs

EXPORT command could not be scheduled.

U0200-00rs

IMPORT command could not be scheduled.

rs

Reason code from the 3495/3494 LCS services. It indicates why the EXPORT/IMPORT command could not be scheduled. The reason codes can be found in the LCS External Services Parameter List (LCSPL), mapped by IBM macro CBRLCSPL. This is used to pass parameters to the LCS external services, to pass return and reason codes and other information back to the caller. This macro can be found in SYS1.MACLIB. U0999 with a dump can be issued if internal logical errors occur.

CTSSYNC ATL RESPONSE LOG Report Sample:

```
MMMDD,YYYY  9:54:12        CTSSYNC ATL RESPONSE LOG
PAGE   1
        FUNCTION ------ATL ACTION --------------STATUS IN TAPE DATABASED
        PROTECT TDI001 SYNC PROCESS COMPLETE    EDM/DYNAM CONTROLLED
        EJECT   TDI002 EJECT COMPLETE
        SCRATCH TDI003 SYNC PROCESS COMPLETE
        PROTECT TDI004 SYNC PROCESS COMPLETE
        PROTECT TDI005 SYNC PROCESS COMPLETE
        PROTECT TDI006 SYNC PROCESS COMPLETE    EDM/DYNAM CONTROLLED
        VERIFY         VERIFY MODE INITIATED
        NOVERIFY       VERIFY MODE TERMINATED
        BULK           BULK-EJECT DOOR ACTIVE
        NOBULK         CONVENIENCE DOOR ACTIVE
        PROTECT TDI016 SYNC PROCESS COMPLETE    EDM/DYNAM CONTROLLED
        PROTECT TDI017 SYNC PROCESS COMPLETE    EDM/DYNAM CONTROLLED
        PROTECT TDI018 SYNC PROCESS COMPLETE    EDM/DYNAM CONTROLLED
        PROTECT TDI019 SYNC PROCESS COMPLETE    VOL NOT IN SCRATCH STATUS
        SCRATCH TDI020 SYNC PROCESS COMPLETE    VOL IN LIBRARY & SCRATCH
        PROTECT TDI021 SYNC PROCESS COMPLETE    EDM/DYNAM CONTROLLED
        PROTECT TDI022 SYNC PROCESS COMPLETE    EDM/DYNAM CONTROLLED
        PROTECT TDI028 SYNC PROCESS COMPLETE    VOL NOT IN SCRATCH STATUS
        PROTECT TDI029 SYNC PROCESS COMPLETE    VOL NOT IN SCRATCH STATUS
        SCRATCH TDI030 SYNC PROCESS COMPLETE    VOL IN LIBRARY & SCRATCH
        SCRATCH TDI031 SYNC PROCESS COMPLETE    VOL IN LIBRARY & SCRATCH
        SCRATCH TDI032 SYNC PROCESS COMPLETE    VOL IN LIBRARY & SCRATCH
ERROR(S)SYNC    ABC001 ATL SYNCRONIZE BYPASSED  VOL NOT DEFINED TO THE
DATABASE
ERROR(S)SYNC    ABC002 ATL SYNCRONIZE BYPASSED  VOL NOT DEFINED TO THE
DATABASE
ERROR(S)SYNC    ABC003 ATL SYNCRONIZE BYPASSED  VOL NOT DEFINED TO THE
DATABASE
ERROR(S)SYNC    ABC004 ATL SYNCRONIZE BYPASSED  VOL NOT DEFINED TO THE
DATABASE
ERROR(S)SYNC    ABC005 ATL SYNCRONIZE BYPASSED  VOL NOT DEFINED TO THE
DATABASE
        SCRATCH TDI001 SYNC PROCESS COMPLETE
        SCRATCH TDI001 VOLUME IS ALREADY IN THAT STATUS
        SCRATCH TDI001 VOLUME NOT FOUND BY ATL
ERROR(S)SCRATCH TDI001 REQUEST FAILED FROM LCS
        IMPORT  TDI003 FROM XMIT10  WRITTEN TO IMPORT LIST VSN MVXE90
        IMPORT  TDI004 FROM XMIT10  WRITTEN TO IMPORT LIST VSN MVXE90
        IMPORT  TDI005 FROM XMIT10  WRITTEN TO IMPORT LIST VSNMVXE90
ERROR(S)EXPORT  TDI006 BYPASSED MIXING IMPORT AND EXPORT
ERROR(S)EXPORT  TDI007 BYPASSED MIXING IMPORT ANDEXPORT
                QUERY   DC0011                  QUERY COMPLETE, VOLUME IN ROBOT
                SYNC    DC0011                  VOLUME WAS ALREADY IN THAT STATUS         VOLUME NOT IN SCRATCH STATUS
                QUERY   DC0012                  QUERY COMPLETE, VOLUME IN ROBOT
                SYNC    DC0012                  VOLUME WAS ALREADY IN THAT STATUS         VOLUME NOT IN SCRATCH STATUS
                QUERY   DC0013                  QUERY COMPLETE, VOLUME IN ROBOT
ERROR(S)        QUERY   DC0013                  OUT OF SYNC: TVIUSEA = S & TVILCAT = UN
                     ---- END OF REPORT ----
```

CTSSYNC report headings and their definitions:

FUNCTION

The function or command or parameter that was passed to CTSSYNC. A SYNC command is broken down first into a QUERY type transaction, to find out whether the cartridge is in the robot and if so what is its status. Then, either a PROTECT command, to change its status from scratch to private or a SCRATCH command, to change its status from private to scratch or a SYNC command, meaning that it was already in the correct status is used.

ATL ACTION

A description of what the ATL has been asked to do and what its response was.

STATUS IN TAPE DATABASE

A description of the status of the cartridge is in the tape management database. This is used only for a SYNC command.

ATL ACTION messages and their meanings:

VERIFY MODE INITIATED

The VERIFY command had been issued and processed.

VERIFY MODE TERMINATED

The NOVERIFY command had been issued and processed.

CONVENIENCE DOOR ACTIVE

The NOBULK command was issued and the convenience door is used for any further EJECT commands.

BULK-EJECT DOOR ACTIVE

The BULK command was issued and the bulk door will be used for any further EJECT commands.

NO DD STATEMENT FOR IMPORT

An IMPORT command was issued, but the JCL did not contain an IMPORT DD statement.

NO DD STATEMENT FOR EXPORT

An EXPORT command was issued, but the JCL did not contain an EXPORT DD statement.

WRITTEN TO EXPORT LIST VSN vvvvvv

An EXPORT command was issued and the volume to be exported was written to the export list created on virtual-volume vvvvvv.

WRITTEN TO IMPORT LIST VSN vvvvvv

An IMPORT command was issued and the volume to be imported was written to the import list created on virtual-volume vvvvvv.

QUERY COMPLETE, VOLUME IN ROBOT

A SYNC command was issued and the volume to be synchronized resides inside the robot.

VOLUME WAS ALREADY IN THAT STATUS

A SYNC command has been issued, and the volume was already in the correct status.

REQUESTED PROCESS COMPLETE

Either a PROTECT, SCRATCH, PRIVATE command had been issued and the volume had its status successfully changed. The PROTECT or SCRATCH command may also have been issued because of a SYNC command when the initial QUERY found the tape was not in correct status.

EXPORT FAILED, REASON CODE=nnn

The EXPORT command failed. The reason codes can be found in the LCS External Services Parameter List (LCSPL), mapped by IBM macro CBRLCSPL. This macro can be found in SYS1.MACLIB.

IMPORT FAILED, REASON CODE=nnn

The IMPORT command failed. The reason codes can be found in the LCS External Services Parameter List (LCSPL), mapped by IBM macro CBRLCSPL. This macro can be found in SYS1.MACLIB.

REQUEST FAILED RC=rc,RS=rsn,FB=nnnnnn

The request (SCRATCH, PRIVATE, PROTECT, SYNC) failed with a return code of rc; and reason code of rsn. The return code and reason code combinations can be found in the LCS External Services Parameter List (LCSPL), mapped by IBM macro CBRLCSPL. This macro can be found in SYS1.MACLIB.

SYNTAX ERROR DURING PARSE

There was a syntax error on the specified control statement.

PARSING ERROR IN THE VSN RANGE

A range of volumes had been specified, but there was a syntax error in the format.

BYPASSED MIXING IMPORT AND EXPORT

It is not allowed to perform both an IMPORT and an EXPORT command on the same execution.

NO COMMAND(S) FOUND

No command was found on the PARM or in the SYSIN.

INVALID REQUEST FOR THIS ATL

ATL software rejected a command as unsupported format.

ATL INACTIVE OR NOT AVAILABLE

The 3494/ATL was either not active or not available to communicate with CTSSYNC.

ATL SYNCHRONIZE BYPASSED

The SYNC command was bypassed for the specified reason.

VOLUME NOT FOUND BY ATL

The volume specified on the command is not defined in the ATL TCDB.

OUT OF SYNC: TVIUSEA = x & TVILCAT = yy

SYNC operation failed. The ATL TCDB status x and LM status yy are different. Use IBM ISPF panels to resolve the discrepancy.

Note: For more information about status settings, see the IBM document *DFSMS Object Access Method Planning, Installation, and Storage Administration Guide for Tape Libraries*.

OUT OF SYNC: ROBOT= xxx & TAPEMGMT= yyy

The tape robot and tape management system both have different settings. This will be fixed if CTSSYNC was run with a PARM='SYNC'.

STATUS IN TAPE DATABASE messages and their meanings:

TAPE FULL OR EDM CONTROLLED

The requested volume was either full or part of a multi-volume chain or controlled by an external data manager. In either case, you can not mod onto it.

VOLUME NOT IN SCRATCH STATUS

The requested volume is active (not in scratch status), but could have additional data MODed onto it or have additional files stacked upon it.

VOLUME IN LIBRARY AND SCRATCH

The requested volume is in scratch status.

VOLUME NOT DEFINED TO THE DATABASE

The volume is not defined to the tape management system.

TAPE MANAGEMENT IS NOT ACTIVE

The tape management system is not currently active.

TAPE CREATED WITHIN THE LAST 5 MINUTES

A SYNC command was issued for a volume that was created within the last 5-minutes; this volume was bypassed.

BAD TAPE OR OUT-OF-SERVICE

A SYNC command was issued for a tape that is either marked as BAD or as OUT-OF-SERVICE in the tape management database.

CTSPMTST Utility—Validate Pattern Definitions

Use the CTSPMTST utility to test the character strings or patterns that you have created for use in the various *hlq*.CTAPOPTN members or utility programs that support pattern masking.

A pattern consists of a combination of alphanumeric characters and special characters that perform unique masking operations during match and compare functions for the purpose of identifying and selecting data sets. The rules for defining a pattern are discussed in the *Programming Guide*.

Once you have created your patterns, test them by executing the CTSPMTST utility. CTSPMTST checks each pattern for conformity to the rules of structure and syntax required by the CA 1 Pattern Masking Facility and produces printed output to aid in resolving invalid pattern errors.

CA 1 can be active, batch active or inactive to run CTSPMTST.

Report Description

CTSPMTST produces hardcopy output which is shown in the sections "Sample Output - Example 1 (see page 53)" and "Sample Output - Example 2 (see page 54)".

Job Control Statements

```
//CTSPMTST EXEC PGM=CTSPMTST,PARM='verb,class'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//SYSPRINT DD  SYSOUT=A
//SYSPATT  DD  *
(control statements here)
/*
//SYSOBJ   DD  *                                           *NOTE
(control statements here)
/*
```

Note: The SYSOBJ DD statement is not required if the verb specified is VALIDATE.

Parameter Definitions

verb

Possible values for verb are:

COMPARE

Compares two patterns and indicates whether the first pattern (SYSPATT DD) is the same, more specific, or less specific than the second pattern (SYSOBJ DD).

VALIDATE

Tests structure and syntax and reports on the validity of the patterns pointed to by the SYSPATT DD. This parameter does not use the SYSOBJ DD.

MATCH

Matches patterns from the SYSPATT DD against objects from the SYSOBJ DD and reports the results.

class

Indicates the object class of the patterns and objects to be processed. These classes define the characteristics of the objects to be matched. Possible values for class are:

MVSFILE

MVS 44-character data set name. This class is nodal and uses the period as a node separator.

MVSJOB

Any standard 8-byte name that can be found in the JCL. This applies to step name, program name, unit name, ddname, and job name. This class is nodeless.

JCL Considerations

SYSPATT DD

Points to one or more patterns used in processing the action requested with the verb parameter. Comment can be used in either the SYSPATT or SYSOBJ file by placing "**" in columns 1 and 2.

SYSOBJ DD

When the COMPARE verb is specified, this DD defines one or more patterns which are compared to the patterns defined in SYSPATT to determine which patterns are more specific.

When the MATCH verb is specified, this DD specifies a list of objects to be matched to the patterns specified in the SYSPATT DD. Each object specified in the SYSOBJ DD is compared to each pattern specified in the SYSPATT DD.

This DD is not used when the VALIDATE verb is specified. It is ignored if provided.

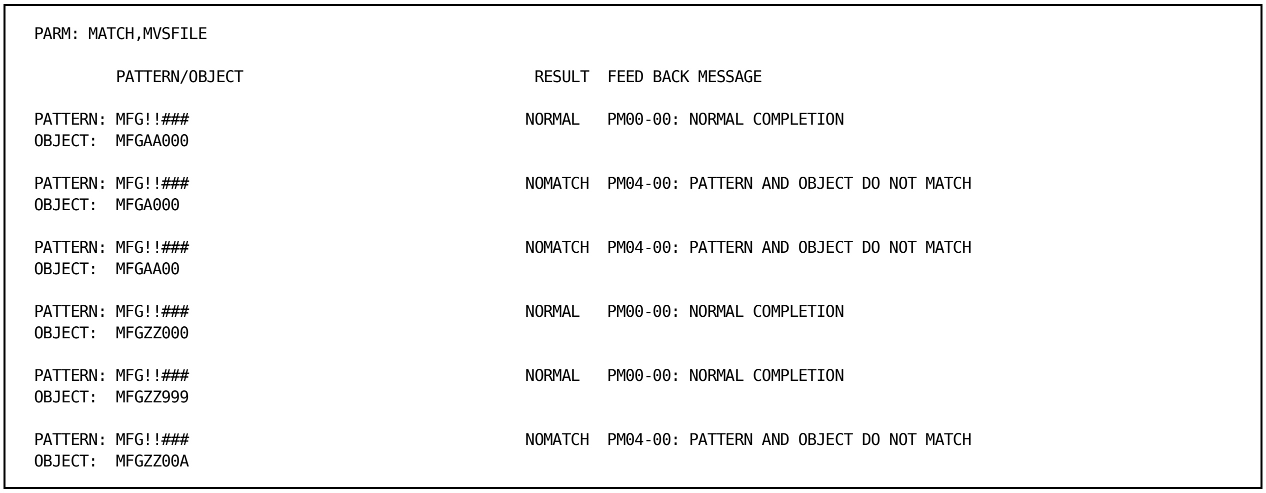
Examples:

Example 1 program names or data set names to identify which EDM is to be assigned. If you plan to use job names to assign the EDM and would like to test the match processing before implementing the rules, execute CTSPMTST with the MATCH,MVSJOB parameter as in the following example. Note that if program name or DD statement were to be tested that the MATCH,MVSJOB parameter would also be used.

```
//STEP1    EXEC PGM=CTSPMTST,PARM='MATCH,MVSJOB'
//STEPLIB  DD DSN=CAI.CTAPLINK,DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSPATT  DD *
MFG!!###
/*
//SYSOBJ   DD *
MFGAA000
MFGA000
MFGAA00
MFGZZ000
MFGZZ999
MFGZZ00A
/*
//
```

Each test job name specified in the SYSOBJ DD is compared to the pattern specified in the SYSPATT DD and the results of the match are reported. The results of executing this job are displayed in Sample Output - Example 1 below.

Sample Output - Example 1



```
PARM: MATCH,MVSFILE

         PATTERN/OBJECT                              RESULT  FEED BACK MESSAGE

PATTERN: MFG!!###                                   NORMAL   PM00-00: NORMAL COMPLETION
OBJECT:  MFGAA000

PATTERN: MFG!!###                                   NOMATCH  PM04-00: PATTERN AND OBJECT DO NOT MATCH
OBJECT:  MFGA000

PATTERN: MFG!!###                                   NOMATCH  PM04-00: PATTERN AND OBJECT DO NOT MATCH
OBJECT:  MFGAA00

PATTERN: MFG!!###                                   NORMAL   PM00-00: NORMAL COMPLETION
OBJECT:  MFGZZ000

PATTERN: MFG!!###                                   NORMAL   PM00-00: NORMAL COMPLETION
OBJECT:  MFGZZ999

PATTERN: MFG!!###                                   NOMATCH  PM04-00: PATTERN AND OBJECT DO NOT MATCH
OBJECT:  MFGZZ00A
```

Example 2

The Data Set Name control statement of the TMSVMEDT Vault utility uses the CA 1 pattern masking facility to determine the data sets to be controlled by VMS. The CTSPMTST program may be used to test this data set selection processing by specifying test vault patterns in the SYSPATT DD and test data set names in the SYSOBJ DD as in the following example.

```
//STEP1    EXEC PGM=CTSPMTST,PARM='MATCH,MVSFILE'
//STEPLIB  DD DSN=CAI.CTAPLINK,DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSPATT  DD *
ABC*.LEVEL2
ABC*.LEVEL#
/*
//SYSOBJ    DD *
ABCDEF.LEVEL3
ABCDEF.LEVEL2
ABCDEF.LEVEL2.ABC
ABC.NOTHING
/*
//
```

The results of executing this job are displayed in the Sample Output - Example 2 below.

Sample Output - Example 2

```
THURSDAY, JUNE 29, 2005.210                      Y O U R   C O M P A N Y   N A M E                                    PAGE=00001

PARM: MATCH,MVSFILE

         PATTERN/OBJECT                                RESULT  FEED BACK MESSAGE

PATTERN: ABC*.LEVEL2                          NOMATCH  PM04-00: PATTERN AND OBJECT DO NOT MATCH
OBJECT:  ABCDEF.LEVEL3

PATTERN: ABC*.LEVEL2                          NORMAL   PM00-00: NORMAL COMPLETION
OBJECT:  ABCDEF.LEVEL2

PATTERN: ABC*.LEVEL2                          NOMATCH  PM04-00: PATTERN AND OBJECT DO NOT MATCH
OBJECT:  ABCDEF.LEVEL2.ABC

PATTERN: ABC*.LEVEL2                          NOMATCH  PM04-00: PATTERN AND OBJECT DO NOT MATCH
OBJECT:  ABC.NOTHING

PATTERN: ABC*.LEVEL#                          NORMAL   PM00-00: NORMAL COMPLETION
OBJECT:  ABCDEF.LEVEL3

PATTERN: ABC*.LEVEL#                          NORMAL   PM00-00: NORMAL COMPLETION
OBJECT:  ABCDEF.LEVEL2

PATTERN: ABC*.LEVEL#                          NOMATCH  PM04-00: PATTERN AND OBJECT DO NOT MATCH
OBJECT:  ABCDEF.LEVEL2.ABC

PATTERN: ABC*.LEVEL#                          NOMATCH  PM04-00: PATTERN AND OBJECT DO NOT MATCH
OBJECT:  ABC.NOTHING
```

CTSPMTST Report Field Definitions

PARM: xxxxxxxx

Parameters passed to program

PATTERN/OBJECT

Pattern or object processed

RESULT

Result of the process

FEED BACK MESSAGE

Processing message

CTSTAPER Utility—Test CA 1 Operating System Intercepts and Functions

Use the CTSTAPER utility to test CA 1 when:

- CA 1 is first installed.
- The operating system release is upgraded.
- CA 1 system options are changed.

CTSTAPER issues OPEN, CLOSE, and FEOV macros to verify the realtime and online system interface. TMSBINQ or the CA 1 online facilities can be used to ensure that activity on the volumes used by CTSTAPER is properly captured for any type of tape data set I/O combination. TMSAUDIT can be used for a detailed analysis of the data management interface.

The program can also be used to verify the console message intercept for scratch subpools, label/density changes, default expiration dates, Auxiliary Disposition or other CA 1 functions.

For tapes to be tracked, CA 1 must be active to run CTSTAPER.

Report Description

CTSTAPER produces the Test Tape OPENs report. A sample of this report appears on page Test Tape OPENs (see page 58).

Job Control Statements

```
//TAPEOUT  EXEC PGM=CTSTAPER[,PARM='[JFCB][,ABEND]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//SYSPRINT DD  SYSOUT=A
//ttnnxxxx DD  DSN=test.dataset.name,UNIT=TAPE,
//             LABEL=EXPDT=99009,DISP=(NEW,CATLG)
//ttnnzzzz DD  ......
//....
```

Parameter Definition

JFCB

Specifies to print the contents in hex of the operating system-constructed JFCB read by CA 1 before OPEN and after OPEN processing. This cannot be specified with ABEND.

ABEND

Specifies to abend (with S0C3) after OPEN to test CA 1 abend processing. This cannot be specified with JFCB.

JCL Considerations

DD names

The ddname determines the type of tape processing to perform. A maximum of nine ddname statements can be specified.

ttnnxxxx

Indicates the type of open, the number of volumes to open, and exclusively identifies the DD statement, where:

tt

Indicates the type of open to be performed:

OU - Open for OUtput

IN - Open for INput

nn

Indicates the number of volumes to be processed.

xxxx

Defines any user-designated characters to make it a unique DD.

Examples:

The following JCL statements will test CA 1 intercepts, multivolume processing, and multi-data set processing, and can be used for both SL and NL tapes. An SL tape test could be:

```
//TAPEOUT  EXEC PGM=CTSTAPER
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//SYSPRINT DD  SYSOUT=A
//OU02SL1  DD  DSN=TEST.FIRST.FILE,DISP=(NEW,CATLG),UNIT=TAPE,
//             VOL=(,RETAIN),LABEL=(1,SL)
//OU01SL2  DD  DSN=TEST.SECOND.FILE,DISP=(NEW,CATLG),UNIT=TAPE,
//             VOL=(,RETAIN,REF=*.OU02SL1),LABEL=(2,SL)
//OU02SL3  DD  DSN=TEST.THIRD.FILE,DISP=(NEW,CATLG),UNIT=TAPE,
//             VOL=(,RETAIN,REF=*.OU01SL2),LABEL=(3,SL)
//OU01SL4  DD  DSN=TEST.FOURTH.FILE,DISP=(NEW,CATLG),UNIT=TAPE,
//             VOL=REF=*.OU02SL3,LABEL=(4,SL)
//TAPEIN   EXEC PGM=CTSTAPER
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//SYSPRINT DD  SYSOUT=A
//IN02SL1  DD  DSN=TEST.FIRST.FILE,DISP=SHR
//IN01SL2  DD  DSN=TEST.SECOND.FILE,DISP=SHR,UNIT=AFF=IN02SL1
//IN02SL3  DD  DSN=TEST.THIRD.FILE,DISP=SHR,UNIT=AFF=IN02SL1
//IN01SL4  DD  DSN=TEST.FOURTH.FILE,DISP=SHR,UNIT=AFF=IN02SL1
```

For nonlabel tapes, specify LABEL=(x,NL) in the above JCL.

Control Statement Specification

There are no control statements for CTSTAPER.

Test Tape OPENs

```
THURSDAY, MMMM DD, YYYY.180                      Y O U R   C O M P A N Y   N A M E                                  PAGE=00001

TEST TAPE OPEN'S

OU01001  ABOUT TO BE OPENED

    JFCB BEFORE     CAI.CA1.TEST.FILE1                                                                   J
                    CCC4CCF4ECEE4CCDCF4444444444444444444444444444444444000000000000008000000000000050C50D0C000000000000
                    319B311B3523B6935100000000000000000000000000000000000000000000000020010000000000A0FA0100000000000000
                   00....5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95....

                                        .
                    000000000000000000F84800444444444444444444444444000000000000000000000000000000
                    0000000000000000001FEB0100000000000000000000000000000000000000000000000000010
                  100....5...10...15...20...25...30...35...40...45...50...55...60...65...70...75

    JFCB AFTER      CAI.CA1.TEST.FILE1                                                                   J
                    CCC4CCF4ECEE4CCDCF4444444444444444444444444444444444000000000000008000000000000050C50D0C000000000040
                    319B311B3523B6935100000000000000000000000000000000000000000000000020010000000000A0FA0100020004000000
                   00....5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95....

                                      980004
                    800101000000000000FFFFFF444444444444444444444444000000000000000800000000000000
                    0008080000000000019800040000000000000000000000000000000000000000000000000010
                  100....5...10...15...20...25...30...35...40...45...50...55...60...65...70...75

OU01001  ABOUT TO BE CLOSED


         END OF REPORT 21  -  TOTAL ITEMS LISTED=         2
```

CTSTAPER Report Field Definitions

JFCB BEFORE

JFCB image before OPEN processing.

JFCB AFTER

JFCB image after OPEN processing.

CTSVDBU Utility - VSAM Database

Use the CTSVDBU utility for the following VSAM database (VDB) batch services:

- List the records in the VDB.
- Initialize VDB control records and CA 1 required variables.
- Add and delete CA 1 required variables after maintenance.

CTSVDBU performs several utility functions against the VDB. The function is specified through the EXEC PARM. You can specify only one function per one execution of CTSVDBU.

The LIST function is probably the most common use of CTSVDBU. You can use the distributed JCL CTSJVDBL for the LIST function. This function writes all VDB records in a field format. The number of fields and their content vary by record type. The function does not write low VDB and high VDB control records. These records do not contain any user information.

The INIT function creates the VPM control records and some variable records that the CA 1 program processing requires. Use the distributed JCL CTSJVDBI for the INIT function.

The UPDATE function creates the same control records and variable records as INIT, but it also deletes obsolete variables. Use the UPDATE function after any CTS maintenance that creates new variables or deletes obsolete variables. To run the UPDATE function, update the distributed INIT JCL CTSJVDBI to specify PARM=’UPDATE’.

Report Description

CTSVDBU produces a list of records and formats the records by type. Each type lists the values of each field it contains. The LIST function displays all the records in the VDB. The INIT and UPDATE functions display only the records that they change.

Job Control Statements

```
//INIT     EXEC PGM=CTSVDBU,PARM='LIST|INIT|UPDATE'
//STEPLIB  DD DSN=CAI.CTAPLINK,DISP=SHR
//*CA$FVDB  DD DSN=CAI.TAPEDB.V,DISP=SHR
//SYSPRINT   DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//[HISTCSV  DD DSN=CAI.HISTCSV,DISP=(NEW,CATLG),
//             RECFM=FB,LRECL=120,BLKSIZE=120,DSORG=PS,
//             UNIT=SYSALLDA,VOL=SER=vvvvvv,SPACE=(TRK,(5,5),RLSE)]
```

Parameter Definition

LIST

Specifies that the CTSVDBU prints the contents of all of the records in the VDB.

INIT

Specifies that CTSVDBU creates all required control records and variables.

UPDATE

Specifies that CTSVDBU creates all required control records and variables. The utility also deletes any obsolete variables.

JCL Considerations

DD CA$FVDB

Only needed when the tape management system has not been started. The VDB is normally dynamically allocated using a DSN that is derived from the TMC by appending a .V to the TMC name. The TMCDSN is available only if the tape system is started.

DD HISTCSV

(Optional) Enables you to create a comma-separated value (CSV) report of the daily and weekly VDB history records when you specify PARM=LIST. Low VDB and high VDB records are not available in this report. You can import this file in any program that edits CSV files.

Control Statement Specification

CTSVDBU does not use any control statements.

CTSVDBU Report

```
20xx/017                    CTSVDBU Utility Report              PAGE=0001

 PARM=LIST

  KEY=@VARCTS_Alert_ID
  CHANGED BY CTSVPM   AT 20xx/017 16:33:06.55 TIMESTAMP=20xx/017 16:33:12.61
  DATA=My_Default_Alert-2

  KEY=@VARCTS_Alert_Operator
  CHANGED BY CTSVPM   AT 20xx/017 16:33:06.55 TIMESTAMP=20xx/017 16:33:12.61
  DATA=GT

  KEY=@VARCTS_Alert_Pool_ID
  CHANGED BY CTSVPM   AT 20xx/017 16:33:06.55 TIMESTAMP=20xx/017 16:33:12.62
  DATA=My_Default_Pool-2

  KEY=@VARCTS_Alert_Send
  CHANGED BY CTSVPM   AT 20xx/017 16:33:06.55 TIMESTAMP=20xx/017 16:33:12.63
  DATA=EMPTYW
```

The report that the CTSVDBU LIST function creates includes multiple record types with different data. The report is not formatted in columns and rows format. The report uses free form data elements with keywords to identify the data that it presents.

CTSVDBBK Utility—Support the Tape Agent Database

Use the CTSVDBBK utility to provide the following services to the Tape Agent database:

BACKUP

Copies the agent database (CTSVDB) to a backup file (CTSVDBB) and then resets the journal file.

RESTORE

Copies the backup file (CTSVDBB) to the database (CTSVDB) and then updates it with the journal entries.

It also provides services to RESET and INIT the journal.

INIT

Formats the journal file. This destroys any entries in that file.

RESET

Sets the journal to the beginning and all existing entries are ignored.

The function to be performed must be specified in the FUNC= keyword of the procedure. Specify 'NULLFILE' for any file that is not to be used.

CTSVDBBK Procedure JCL

The following procedure may be used for all functions of the VDB utility.

```
//******************************************************************
//* THIS JOB WILL PERFORM VDB RELATED FUNCTIONS.                   *
//*    FUNC = BACKUP   = BACKUP VDB & RESET JOURNAL                *
//*         = RESTORE  = CREATES VDB FROM BACKUP AND JOURNAL       *
//*         = RESET    = RESET JOURNAL                             *
//*         = INIT     = INITIALIZE JOURNAL                        *
//******************************************************************
//*
//VDBUTY    PROC SOUT='*',
//              LIB='CAI.CTAPLINK',
//              VDB='CAI.CA$FVDB',
//              VDBB='CAI.CA$FVDBB',
//              JRNL='CAI.CAIJRNL',
//              FUNC='BACKUP '
//*
//VDBU      EXEC PGM=CTSVDBBK,PARM='&FUNC'.
//STEPLIB   DD   DISP=SHR,DSN=&LIB.
//CAIJRNL   DD   DISP=SHR,DSN=&JRNL.
//CA$FVDB   DD   DISP=SHR,DSN=&VDB.
//CA$FVDBB  DD   DISP=SHR,DSN=&VDBB.
//SYSPRINT  DD   SYSOUT=&SOUT.
//SYSUDUMP  DD   SYSOUT=&SOUT.
//*
```

Starter JCL

JCL to backup the agent database

```
//VDBJOB    JOB
//VDBSTP    EXEC FUNC='BACKUP '
//
```

JCL to restore the agent database

```
//VDBJOB    JOB
//VDBSTP    EXEC FUNC='RESTORE'
//
```

JCL to reset the journal

```
//VDBJOB    JOB
//VDBSTP    EXEC FUNC='RESET  '
//
```

JCL to format the journal

```
//VDBJOB    JOB
//VDBSTP    EXEC FUNC='INIT   '
//
```

CTSVDBBK Output

The backup utility has four functions:

- BACKUP
- RESTORE
- RESET
- INIT

The following reports represent each function.

BACKUP REPORT

```
LOGGED: YYYY/316 11.57.05              CA-COMMON TAPE SYSTEM MESSAGE LOG             PAGE  1
CTSVDBBK - BACKUP
CTSVDBBK - BACKUP  START - YYYY/316  115705
CTSVDBBK - VDB--IN VDB-OUT BKU--IN BKU-OUT JRN--IN JRN-OUT
CTSVDBBK -    8     1    0    7     1    1
CTSVDBBK - BACKUP  STOP - YYYY/316  115706
```

RESTORE REPORT

```
LOGGED: YYYY/316 12.07.56              CA-COMMON TAPE SYSTEM MESSAGE LOG             PAGE  1
CTSVDBBK - RESTORE
CTSVDBBK - RESTORE  START - YYYY/316  120756
CTSVDBBK - VDB--IN VDB-OUT BKU--IN BKU-OUT JRN--IN JRN-OUT
CTSVDBBK -    0     7    7    0     1    0
CTSVDBBK - RESTORE  STOP - YYYY/316  120756
```

RESET REPORT

```
LOGGED: YYYY/316 12.07.18              CA-COMMON TAPE SYSTEM MESSAGE LOG             PAGE  1
CTSVDBBK - RESET
CTSVDBBK - RESET  START - YYYY/316  120718
CTSVDBBK - VDB--IN VDB-OUT BKU--IN BKU-OUT JRN--IN JRN-OUT
CTSVDBBK -    0     0    0    0     1    1
CTSVDBBK - RESET   STOP - YYYY/316  120718
```

INIT REPORT

```
LOGGED: YYYY/316 12.07.34              CA-COMMON TAPE SYSTEM MESSAGE LOG             PAGE  1
CTSVDBBK - INIT
CTSVDBBK - INIT  START - YYYY/316  120734
CTSVDBBK - VDB--IN VDB-OUT BKU--IN BKU-OUT JRN--IN JRN-OUT
CTSVDBBK -    0     0    0    0     1   360
CTSVDBBK - INIT   STOP - YYYY/316  120734
```

CTSXLIST Utility - Export Virtual Volumes onto Physical Volumes

Use the CTSXLIST utility to read volsers from a VTS Export Status output file from CTSSYNC after a COPY/Export (COPY,POOL=xx) operation has completed. This utility reads the status file (file-3) on the tape that CTSSYNC created as input. The utility then drives the standard CBRUXEJC exit as if a MOVE/Export operation has been performed. This action updates the ACTVOL field within the tape management system to indicate which physical volume the virtual volumes have been stacked upon.

Run this utility after you performed Copy/Export processing using the CTSSYNC utility.

Report Description

The CTSXLIST utility produces the CTSXLIST report. (see page 65)

Job Control Statements

```
//stepname EXEC PGM=CTSXLIST[,PARM='TEST']
//STEPLIB  DD DSN=CAI.CTAPLINK,DISP=SHR
//INPUT    DD DSN=ctssync.output.file,DISP=SHR,        *
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//CTSRPT   DD SYSOUT=A
//CTSRPT2  DD SYSOUT=A
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
```

Control Statement Specification

CTSXLIST does not have any control statements.

Parameter Definitions

TEST

Process normally but do not eject selected tapes.

CTSXLIST Report

```
WEDNESDAY, MAY 16, 2012.137    COMPLETE    JOB=XLIST   STEP=STEP1   TIME=05:50:28
PAGE=00001
                                             CTSXLIST REPORT
     V-VOLSER  P-VOLSER  EXPORT RC  CBRUXEJC CALLED
     ========  ========  =========  ===============
      400063    730225      00            YES
      400138    730225      00            YES
      400192    730225      00            YES
      400200    730225      00            YES
      400424    730225      00            YES
      400466    730225      00            YES
      400498    730225      00            YES
      400529    730225      00            YES
      400542    730225      00            YES
      400754    730225      00            YES
         END OF REPORT COMPLETE    -  TOTAL ITEMS LISTED=       10
```

CTSXLIST Report Field Definitions

V-VOLSER

Virtual volume serial number that is exported to new physical volume.

P-VOLSER

Physical volume serial number that received the virtual volume.

EXPORT RC

Return code that is read from the export output file.

CBRUXEJC CALLED

IBM Volume Eject exit CBRUXEJC called to eject this volume.

DYNCOMM Utility—Support CA Dynam/T Interface Queue Processing

Use the DYNCOMM utility to send and receive transaction records of all tape activity between CA 1 and CA Dynam/T.

All tapes used by the CA Dynam/T system must be initially defined in the CA 1 Tape Management Catalog (TMC).

Information is extracted from the CA 1 Audit data set and written to the common communications queue to be used to update the CA Dynam/T catalog; conversely, CA Dynam/T tape activity information is updated in the CA 1 TMC from a common communication queue.

DYNCOMM can be executed either as a started task or batch program. If DYNCOMM is to be executed as a started task, the MVS console operator must initiate its execution by issuing the following MVS start command:

```
S DYNCOMM
```

When executed, DYNCOMM attaches two subtasks, SEND and RECV. If there is no data to send (there are no new records on the Audit data set), the SEND task will wait for a user-specified period of time. If there are no records to receive, the RECV task will turn itself off for a user-specified period. When the SEND task again becomes active, it begins processing the Audit data set at the point where it ceased processing during the prior period of activity.

When records are sent from CA 1, they are placed on a DASD queue (the *sending* queue), which is shared with CA Dynam/T. Likewise, when records are received, they are read from a different DASD queue (the *receiving* queue), which is also shared with CA Dynam/T. The CA 1 sending queue is the same data set that CA Dynam/T uses as its receiving queue; the CA 1 receiving queue is the same data set as the CA Dynam/T sending queue. This queue relationship is graphically illustrated by the figure on DYNCOMM - CA Dynam/T Interface Queue Processing (see page 65).

Once started, DYNCOMM continues to execute until the MVS console operator issues the following MVS stop command:

```
P DYNCOMM
```

CA 1 must be active or batch active to run DYNCOMM.

Report Description

DYNCOMM produces a Detail Log file and an Error Log file. The Detail Log file logs one record for each CA Dynam/T transaction, thereby recording all activity associated with the interface including SEND and RECV transactions, start and stop activity, and all error transactions. The Error Log file is identical in form to the Detail Log file, but contains only error transactions (records that were not processed).

The Detail Log file and the Error Log file can be output to a data set (disk or tape) or to SYSOUT. The DYNCOMM task must be stopped to process or print the files. A suggested method is to put the Detail Log file to SYSOUT and the Error Log file to a GDG disk data set.

The Detail Log file is needed to resolve any queuing problems or to examine events thoroughly after a problem has occurred to resolve any questions regarding the actual sequence of events. The Error Log file should be examined to ensure that CA 1 and CA Dynam/T stay in synchronization with each other at all times. Samples of the Detail Log file and Error Log file appear on Example (see page 69).

Job Control Statements

```
//stepname     EXEC PGM=DYNCOMM[,PARM='DATEFMT=(fmt)']
//STEPLIB      DD DSN=CAI.CTAPLINK,DISP=SHR
//AUDIT        DD DSN=CAI.CA1.AUDIT,DISP=SHR
//ca12dyn      DD DSN=cai.ca12dyn.DISP=(NEW,CATLG,DELETE)
//dyn2ca1      DD DSN=cai.dyn2ca1,DISP=(NEW,CATLG,DELETE)
//ERRORLOG     DD SYSOUT=A,DCB=BLKSIZE=130
//DETAILOG     DD SYSOUT=A,DCB=BLKSIZE=130
//SYSIN        DD DSN=hlq.CTAPOPTN(TMODYNxx),DISP=SHR
```

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 20. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

The following CA Dynam/T Interface options must be specified in the TMODYNxx member of *hlq*.CTAPOPTN:

TYPE=

TYPE= is required and must be specified as one of the following:

WARM

Start at the point where processing last stopped.

COLD

Start at the current time. Do not process any old records.

BACK

Start at the point in processing when the last TMSCOPY backup was completed.

SPATHID=ca12dyn

The one- to eight-character identifying name of the sending path; also the name of the DD statement pointing to the sending queue. SPATHID is required.

RPATHID=dyn2ca1

The one- to eight-character identifying name of the receiving path; also the name of the DD statement pointing to the receiving queue. RPATHID is required.

CPU=x

A one-character identification used to indicate the system holding the lock on the queues. (It is identical to the CPUID specified in the XSYSTEM macro.) CPU is required.

SWAIT={nnnn|60}

An optional parameter specified in up to four digits (range 5-1800) designating, in seconds, how long the sending task should wait before checking the Audit data set for more records to send. The default is 60.

RWAIT={nnnn|60}

An optional parameter specified in up to four digits (range 5-1800) designating, in seconds, how long the receiving task should wait before checking the receiving queue for more records to process. If unspecified, a default of 60 (seconds) is assumed.

RVERIFY={nn|10}

An optional parameter specified in up to two digits (range 5-99), indicating to the receiving task the number of wait periods to expire, without receiving data, before sending a verify transaction (to ensure that communications are still established). If unspecified, a default of 10 (wait periods) is assumed.

The time period produced is calculated to be RWAIT times the number specified here.

Example: :

```
RWAIT=120 (seconds) x RVERIFY=5 (wait periods) = 600 seconds = 10 minutes
```

WVERIFY={nn|5}

An optional parameter specified in up to two digits (range 5-99), indicating to the receiving task the number of wait periods to expire, without receiving any response to the verify request, before issuing the IEBDYN20 message. If unspecified, a default of 5 (wait periods) is assumed.

The time period produced is calculated to be RWAIT times the number specified here.

PER={nn|80}

An optional parameter specified in up to two digits (range 50-99), indicating the percentage of the sending queue used, before issuing the IEBDYN21 message. This warning message serves notification that the sending queue is filling up. If unspecified, a default of 80 percent is assumed.

Example

To illustrate a typical use of the DYNCOMM parameters, assume that the following are in effect:

```
TYPE=WARM
SPATHID=CA12DYN
RPATHID=DYN2CA1
CPU=Z
SWAIT=120
RWAIT=120
RVERIFY=15
```

Both SEND and RECV will wait for 2 minutes (120 seconds) before checking (either the Audit data set or receiving queue) for more transactions to process. If the receiving task does not receive any transactions for 30 minutes (120 seconds x 15), a verify transaction is sent. If the verify transaction is not returned within 10 minutes (120 seconds x 5, the WVERIFY default), the IEBDYN20 message is issued.

If the sending queue becomes 80 percent full (the default), the IEBDYN21 message is issued.

```
  THE COMMUNICATION PROGRAM FOR THE CA 1. CA DYNAM/T INTERFACE WAS STARTED ON MMMDD YYYY AT 18:50   DETAIL LOG

 SEND  MMMDD YYYY 18:50          SEND TASK HAS STARTED                        NORM
 RECV  MMMDD YYYY 18:50          RECEIVE TASK HAS STARTED                     NORM
 RECV  MMMDD YYYY 18:50  000009  DYNT.DATASET.TWJRK                           NORM                      MMMDD YYYY  1841
 RECV  MMMDD YYYY 18:50  000002  DYNT.TEST.TAPE                               NORM                      MMMDD YYYY  1841
 SEND  MMMDD YYYY 18:50  980006  USERID1.TEST.SINGLE.FILE                     NORM   000000358       OUTPUT
 SEND  MMMDD YYYY 18:50  055239  TAPE.WENT.SCRATCH                            NORM   000000555       SCRTCH
 SEND  MMMDD YYYY 18:50  055380  TAPE.WENT.SCRATCH                            NORM   000000556       SCRTCH
 SEND  MMMDD YYYY 18:51  055380  UPDATED.BY.TSO                               NORM   000000557       BATCH
 SEND  MMMDD YYYY 18:51  057403  SMPMCS                                       THOLD  000000607       INPUT
 SEND  MMMDD YYYY 18:51  055313  I8508808.SMPMCS                              THOLD  000000611       OUTPUT
 RECV  MMMDD YYYY 18:51          RECORD TYPE FOUND= 3                         ERROR  BAD RECORD TYPE 04
 RECV  MMMDD YYYY 18:51  000005  ADMIN01.CA01.DATASET.D005                    ERROR  NON-DYNAM TAPE  1C MMMDD YYYY  1200
 RECV  MMMDD YYYY 18:51  000010  DYNT.DATASET.D010                            NORM                      MMMDD YYYY  1316
 RECV  MMMDD YYYY 18:52  000005  ADMIN01.CA01.DATASET.D006                    ERROR  NO-MATCH DSN    10 MMMDD YYYY  1200
 RECV  MMMDD YYYY 18:52  000005  ADMIN01.CA01.DATASET.D005                    ERROR  NO-MATCH CRTTI  18 MMMDD YYYY  1200
 RECV  MMMDD YYYY 18:52  000005  ADMIN01.CA01.DATASET.D005                    ERROR  NO-MATCH CRTDT  14 MMMDD YYYY  1200
 RECV  MMMDD YYYY 18:52  000005  ADMIN01.CA01.DATASET.D005                    NORM                      MMMDD YYYY  1200
 RECV  MMMDD YYYY 18:52  000009  TAPE.WENT.SCRATCH                            NORM                      MMMDD YYYY  1200
 RECV  MMMDD YYYY 18:53  000005  TAPE.WENT.SCRATCH                            ERROR  NON-DYNAM TAPE  20 MMMDD YYYY  1201
 RECV  MMMDD YYYY 18:53  000010  DYNT.DATASET.D010                            NORM                      MMMDD YYYY  1316
 RECV  MMMDD YYYY 18:53  000005  UPDATED.BY.TSO                               ERROR  NON-DYNAM TAPE  0C MMMDD YYYY  1201
 RECV  MMMDD YYYY 18:53  056082  CAAM.SMPMCS                                  ERROR  INVALID VOLSER  08 MMMDD YYYY  1733
 RECV  MMMDD YYYY 18:54  060941  RPT.ARCLOG                                   ERROR  INVALID VOLSER  08 MMMDD YYYY  1811
 SEND  MMMDD YYYY 15:54  057403  SMPMCS                                       FULL   000000613       INPUT
 RECV  MMMDD YYYY 18:54  VERIFY  DYNAM2CA  WAS PASSED TO SEND                 NORM
 RECV  MMMDD YYYY 18:58  VERIFY  DYNAM2CA  WAS NOT RETURNED                   ERROR
 SEND  MMMDD YYYY 18:58  VERIFY  DYNAM2CA  WAS SENT                           NORM
 RECV  MMMDD YYYY 18:58  VERIFY  DYNAM2CA  WAS RECEIVED                       NORM
 REVC  MMMDD YYYY 18:59  08  18  RECEIVE TASK HAS TERMINATED                  FAIL
 SEND  MMMDD YYYY 18:59          SEND TASK HAS TERMINATED                     NORM

|____||_______________||______||___________________________________________||_____||_____________||_________________|
  |           |            |                        |                           |          |                  |
  |           |            |                        |                           |          |                  |
  1           2            3                        4                           5          6                  7
```

Detail Log File

```
 THE COMMUNICATION PROGRAM FOR THE CA 1. DYNAM/T INTERFACE WAS STARTED ON MMMDD YYYY AT 18:50   ERROR LOG

 RECV  MMMDD YYYY 18:51         RECORD TYPE FOUND= 3                         ERROR  BAD RECORD TYPE 04
 RECV  MMMDD YYYY 18:51  000005  ADMIN01.CA01.DATASET.D005                    ERROR  NON-DYNAM TAPE  1C MMMDD YYYY  1200
 RECV  MMMDD YYYY 18:52  000005  ADMIN01.CA01.DATASET.D006                    ERROR  NO-MATCH DSN    10 MMMDD YYYY  1200
 RECV  MMMDD YYYY 18:52  000005  ADMIN01.CA01.DATASET.D005                    ERROR  NO-MATCH CRTTI  18 MMMDD YYYY  1200
 RECV  MMMDD YYYY 18:52  000005  ADMIN01.CA01.DATASET.D005                    ERROR  NO-MATCH CRTDT  14 MMMDD YYYY  1200
 RECV  MMMDD YYYY 18:53  000005  TAPE.WENT.SCRATCH                            ERROR  NON-DYNAM TAPE  20 MMMDD YYYY  1201
 RECV  MMMDD YYYY 18:54  000005  UPDATED.BY.TSO                               ERROR  NON-DYNAM TAPE  0C MMMDD YYYY  1201
 RECV  MMMDD YYYY 18:54  056082  CAAM.SMPMCS                                  ERROR  INVALID VOLSER  08 MMMDD YYYY  1733
 RECV  MMMDD YYYY 18:54  060941  RPT.ARCLOG                                   ERROR  INVALID VOLSER  08 MMMDD YYYY  1811
 SEND  MMMDD YYYY 15:54  057403  SMPMCS                                       FULL   000000613       INPUT
 RECV  MMMDD YYYY 15:58  VERIFY  DYNAM2CA  WAS NOT RETURNED                   ERROR
 RECV  MMMDD YYYY 15:59  08  18  RECEIVE TASK HAS TERMINATED                  FAIL

|____||________________||______||___________________________________________||_____||______________||___________________|
  |           |            |                        |                           |          |                  |
  |           |            |                        |                           |          |                  |
  1           2            3                        4                           5          6                  7
```

Error Log File

Detail Log and Error Log Data Definitions

The field numbers below correspond with those indicated at the bottom of the report samples on the previous pages.

1

The function that processed the transaction, either SEND or RECV.

2

The date and time that the transaction was processed.

3

The volume serial number. If this was a verify transaction, this field contains the word VERIFY.

If the SEND or RECV task has failed, this field contains the error return code and reason code. See Field 5 definition for FAIL return codes.

4

The data set name if a VOLSER is present. This field can contain SEND/RECV task or error information.

5

The transaction status, as follows:

NORM - Normal transaction; processed successfully.

THOLD - Transaction occurred after sending queue reached percent full threshold; processed successfully.

FULL - Transaction occurred after the sending queue was full. The transaction was not processed. A record has been written to the Error Log file.

ERROR - A bad transaction was received. A record has been written to the Error Log file. See Field 7 definition for ERROR return codes.

LFAIL - DYNCOMM was unable to obtain the lock for the Send or Receive queue. The operator is asked to retry the lock or cancel DYNCOMM.

Note: For more information about this failure, see the *Message Reference Guide*, messages IEBDYN23 and IEBDYN24.

FAIL - A task has ended with an error termination. The return code and reason code appear in the VOLSER field. A record has been written to the Error Log file. FAIL return code definitions are:

04

An invalid request has been made against a queue. This return code represents an internal problem with the program making the request. For assistance in resolving this problem, contact the appropriate Computer Associates representative for the system under which you are operating the interface (CA 1 or CA Dynam/T).

08

The PATHID for the queue specified does not match the identification in the queue header read.

24

A critical error has occurred within the queue processor (possibly a storage overlay). For assistance in resolving this problem, obtain a dump and Detail and Error Logs and contact the appropriate CA representative for the system under which you are operating the interface (CA 1 or CA Dynam/T).

28

The CA 1 Audit data set has been destroyed. See the TMSFORMT discussion in this manual or the *Programming Guide* for instructions on creating a new Audit data set.

32

I/O error reading the Audit data set. Verify that the JCL points to the correct Audit data set. Verify that the Audit data set has not been destroyed and that it is properly formatted.

36

The CA 1 Audit data set did not open successfully. Ensure that the correct data set is referenced by the AUDIT DD statement.

6

If a SEND transaction, this field contains the number of the Audit data set record that was sent. If an error occurred on an RECV transaction, this field contains a description of the error.

7

For a SEND transaction, the type of processing performed, as follows:

OUTPUT - Opened for output.
INPUT - Opened for input.

BATCH - Accessed by a batch job.
SCRTCH - Tape flagged for scratch.

If an error occurred on an RECV transaction, this field contains the error return code, and the creation date and time from the CA Dynam/T record. Return code definitions are:

04

Bad record type from CA Dynam/T. Contact CA Dynam/T Technical Support.

08

INVALID VOLSER - The VOLSER was not defined in the CA 1 TMC.

0C

NON-DYNAM TAPE - A manual update was received for a tape not controlled by CA Dynam/T.

10

NO-MATCH DSN - A tape controlled by CA 1 was used by CA Dynam/T for input processing, but with the wrong data set name.

14

NO-MATCH CRTDT - A tape controlled by CA 1 was used by CA Dynam/T for input processing, but with the wrong creation date.

18

NO-MATCH CRTTI - A tape controlled by CA 1 was used by CA Dynam/T for input processing, but with the wrong creation time.

1C

NON-DYNAM TAPE - CA Dynam/T processed a CA 1 controlled tape for output.

20

NON-DYNAM TAPE - CA Dynam/T placed into SCRATCH status a tape not controlled by CA Dynam/T.

24

DELETED TAPE - CA Dynam/T processed for output a tape in DELETE status.

DYNEXTRA Utility—Build the Database Extract for DYNSYNC

Use the DYNEXTRA utility to build the database extract from the TMC data set.

The resulting extract is processed by DYNSYNC, the database synchronization utility. This utility does not produce a report.

CA 1 should be deactivated before executing DYNEXTRA to prevent updates to the existing TMC. If output is to tape, a nonresident volume should be used (one not currently defined in the TMC).

CA 1 can be active, batch active or inactive to run DYNEXTRA.

Report Description

DYNEXTRA does not generate any reports.

Job Control Statements

```
//stepname EXEC PGM=DYNEXTRA
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMC      DD  DSN=CAI.CA1.TMC,DISP=SHR
//EXTRACT  DD  DSN=CAI.CA1.EXTRACT.DSN,DISP=(NEW,CATLG,DELETE),
//             SPACE=(CYL,(10,2)),UNIT=SYSDA,
//             DCB=(RECFM=FB,LRECL=216,BLKSIZE=n*216)
//SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
//SYSOUT   DD  SYSOUT=A
//SORTWK01 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK02 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK06 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
```

Space requirements for this utility are calculated by determining how many records are generated.This number is equal to the number of volumes defined +1 (Control Record).

DYNFORMT Utility—Format CA 1 CA Dynam/T Interface Queues

Use the DYNFORMT utility to format a queue so that it can be used by the communications interface.

DYNFORMT is capable of formatting up to two queues at a time. The PATH name and queue size must be supplied for each queue to be formatted. The PATH name corresponds to the ddname pointing to the queue to be formatted.

CA 1 can be active, batch active or inactive to run DYNFORMT.

Report Description

DYNFORMT produces TMS Report 73 - CA 1 CA Dynam/T Interface Queue Format Report. This report identifies the queue(s) that has been formatted, and the total number of transaction records that this queue can contain before it is considered full; this total is equivalent to the queue size. A sample of this report appears on page TMS Report 73 - CA 1 CA Dynam/T Interface Queue Format Report (see page 75).

Job Control Statements

```
//stepname EXEC PGM=DYNFORMT,
//             PARM='SPATHID=ca12dyn,SQSIZE=nnnnnn,
//             RPATHID=dyn2ca1,RQSIZE=nnnnnn'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//ca12dyn  DD  DSN=cai.ca12dyn.DISP=(NEW,CATLG,DELETE),
//             UNIT=SYSDA,SPACE=(CYL,(10,2)),
//             DCB=(RECFM=FB,LRECL=216,BLKSIZE=216)          *NOTE
//dyn2ca1  DD  DSN=cai.dyn2ca1,DISP=(NEW,CATLG,DELETE),
//             UNIT=SYSDA,SPACE=(CYL,(10,2)),
//             DCB=(RECFM=FB,LRECL=216,BLKSIZE=216)          *NOTE
```

Note: BLKSIZE must be 216.

SPATHID and SQSIZE relate to one queue and, if specified, RPATHID and RQSIZE represent the other queue. SPATHID and RPATHID specify the path name(s) to be placed in the formatted queue(s) and must be valid ddnames. SQSIZE and RQSIZE specify the number of records the queue(s) can contain, including one Control Record for each. SQSIZE and RQSIZE can be up to six digits and must be numeric.

TMS Report 73 - CA 1 CA Dynam/T Interface Queue Format Report

```
DATES=04/01/95.091                              Y O U R   C O M P A N Y   N A M E                                PAGE=00001

TMS REPORT-73  CA 1 CA DYNAM/T INTERFACE QUEUE FORMAT REPORT

QUEUE CA12DYN  FORMATTED WITH    2000 RECORDS
QUEUE DYN2CA1  FORMATTED WITH    2000 RECORDS
```

DYNRELSE Utility—Release CA Dynam/T Interface Queues

Use the DYNRELSE utility to release (free) a queue locked as a result of a system failure.

If the system fails while the queue is locked, the queue will remain locked and inaccessible to all queuing routines.

DYNRELSE is designed to be run as a started task from the MVS operator console. It can also be run as a batch job. The queue to be released is passed as a parameter to the DYNRELSE routine. One or both queues can be passed as parameters to DYNRELSE to be released.

Note: All CA 1 and CA Dynam/T communications (DYNCOMM) tasks must be stopped before executing this utility. The MVS console operator must issue the following MVS stop command:

```
P DYNCOMM
```

To resume communications, the MVS console operator must issue the following MVS start command:

```
S DYNCOMM
```

CA 1 can be active, batch active or inactive to run DYNRELSE.

Report Description

DYNRELSE does not generate any reports.

Job Control Statements

```
//stepname EXEC PGM=DYNRELSE,PARM='SPATHID=ca12dyn,RPATHID=dyn2ca1'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//ca12dyn  DD  DSN=cai.ca12dyn.DISP=(NEW,CATLG,DELETE),
//dyn2ca1  DD  DSN=cai.dyn2ca1,DISP=(NEW,CATLG,DELETE),
```

SPATHID and RPATHID correspond to the PATHID as specified in the XSYSTEM macro under CA Dynam/T.

All output from DYNRELSE is sent to the MVS operator console. The messages indicate the actions taken by DYNRELSE.

DYNSYNC Utility—Synchronize the CA Dynam/T Databases

Use the DYNSYNC utility to compare the database extracts from each system. DYNEXTRA generates the database extract for CA 1. A DOS version of this utility is available from CA Dynam/T under the name, TDYNSYNC.

An output report is generated indicating the discrepancies between the two databases. Additional output consists of data for the CA 1 and CA Dynam/T update programs for the respective databases.

CA 1 can be active, batch active or inactive to run DYNSYNC.

Report Description

DYNSYNC produces TMS Report 70 - Tape Management Database Synchronization. A sample of this report appears on page TMS Report 70 - Tape Management Database Synchronization Report (see page 77).

Job Control Statements

```
//stepname EXEC PGM=DYNSYNC
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//SYS1IN   DD  DSN=cai.ca1.extract(or)cai.cadynamt.extract,DISP=SHR
//SYS2IN   DD  DSN=cai.ca1.extract(or)cai.cadynamt.extract,DISP=SHR
//SYS1OUT  DD  DSN=cai.ca1.syncdata(or)cai.cadynamt.syncdata,
//             DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(10,2)),UNIT=SYSDA,
//             DCB=(RECFM=FB,LRECL=216,BLKSIZE=n*216)
//SYS2OUT  DD  DSN=cai.ca1.syncdata(or)cai.cadynamt.syncdata,
//             DISP=(NEW,CATLG,DELETE),SPACE=(CYL,(10,2)),UNIT=SYSDA,
//             DCB=(RECFM=FB,LRECL=216,BLKSIZE=n*216)
```

SYS1IN and SYS2IN can point to the extract from either database. SYS1OUT and SYS2OUT will contain the synchronization information for the systems represented by SYS1IN and SYS2IN respectively.

TMS Report 70 - Tape Management Database Synchronization Report

```
DATES=04/01/95.091                               Y O U R   C O M P A N Y   N A M E                                  PAGE=00001

TMS REPORT-70  TAPE MANAGEMENT DATA BASE SYNCHRONIZATION

                       CA 1/MVS                                  DYNAM/T
VOLSER   DATA SET NAME  STAT   CREATED   SYSTEM    DATA SET NAME  STAT   CREATED  SYSTEM  DEST MESSAGE

000002 CA01.DATASET.D002 SCR 95002 1200          CA01.DATASET.D002 ACT 95002 1200 CA1MVS  SYS2 VOLUME STATUS DOES NOT MATCH
000003 DYNT.DATASET.D003 SCR 95003 1200          DYNT.DATASET.D003 ACT 95003 1200 DYNAMT  SYS1 VOLUME STATUS DOES NOT MATCH
000004 CA01.DATASET.D004 SCR 95004 1200                                                   SYS2 VOLSER NOT ON SYS2 EXTRACT
000005 CA01.DATASET.D005 ACT 95005 1200 CA1MVS   CA01.DATASET.D005     95005 1200         SYS2 VOLUME STATUS DOES NOT MATCH
000007 CA01.DATASET.D007 ACT 95007 1200 CA1MVS   DYNT.DATASET.D007 ACT 95007 1200 DYNAMT  SYS1 EACH SYSTEM CLAIMS OWNERSHIP
000008 CA01.DATASET.D008 ACT 95008 1200 CA1MVS                                            SYS2 VOLSER NOT ON SYS2 EXTRACT
000009 DYNT.DATASET.D009 ACT 95009 1200 DYNAMT   DYNT.DATASET.D009 SCR 95009 1200         SYS1 VOLUME STATUS DOES NOT MATCH
000010 DYNT.DATASET.D010 ACT 95010 1200 DYNAMT   CA01.DATASET.D010 ACT 95010 1200 CA1MVS  BOTH NEITHER SYSTEM CLAIMS OWNERSHIP
000012 DYNT.DATASET.D012 ACT 95012 1200 DYNAMT                                            SYS2 VOLSER NOT ON SYS2 EXTRACT
000013                                           DYNT.DATASET.D013 SCR 95013 1200         SYS1 VOLSER NOT ON SYS1 EXTRACT
000014                                           CA01.DATASET.D014 ACT 95014 1200 CA1MVS  SYS1 VOLSER NOT ON SYS1 EXTRACT
000015                                           DYNT.DATASET.D015 ACT 95015 1200 DYNAMT  SYS1 VOLSER NOT ON SYS1 EXTRACT
000017 CA01.DATASET.ALV3 ACT 95017 1200 CA1MVS                     SCR 00000 0000         SYS2 VOLUME STATUS DOES NOT MATCH
000018 CA01.DATASET.ALV3 ACT 95018 1200 CA1MVS                     SCR 00000 0000         SYS2 VOLUME STATUS DOES NOT MATCH
000019 CA01.DATASET.D019 ACT 95019 1200 CA1MVS   CA01.DATASET.D0XX ACT 95019 1200 CA1MVS  SYS2 DATA SET NAMES DO NOT MATCH
000020 CA01.DATASET.D020 ACT 95020 1200 CA1MVS   CA01.DATASET.D020 ACT 95020 1200 CA1MVS  SYS2 CREATION DATE/TIME DOES NOT MATCH
000021 DYNT.DATASET.D021 ACT 95021 1200 DYNAMTA  DYNT.DATASET.D021 ACT 95021 1200 DYNAMT  SYS1 SYSTEM NODEIDS DO NOT MATCH
000101                                                             SCR 00000 0000         SYS1 VOLSER NOT ON SYS1 EXTRACT
000102                                                             SCR 00000 0000         SYS1 VOLSER NOT ON SYS1 EXTRACT
000103                                                             SCR 00000 0000         SYS1 VOLSER NOT ON SYS1 EXTRACT
000104                                                             SCR 00000 0000         SYS1 VOLSER NOT ON SYS1 EXTRACT
000105                                                             SCR 00000 0000         SYS1 VOLSER NOT ON SYS1 EXTRACT

                                    SYS1      SYS2
          TOTAL RECORDS READ          97       102
          TOTAL RECORDS WRITTEN       14        11
          TOTAL RECORDS INVALID        0         0
```

DYNSYNC Report Field Definitions

VOLSER

This field represents the volume in error.

DATA SET NAME

This field contains the last 17 characters of the data set name for the volume being compared. For ANSI Version 3 labeled tapes, this is the last 17 characters before the generation number (GDG). This field may be blank for volumes in a multivolume multi-data set group, or for scratch tapes that have never been used.

STAT

This field indicates the status of the volume on the system it is listed under. The value SCR represents volumes in scratch status; ACT represents volumes that are active (in nonscratch status).

CREATED

These fields represent the creation date and time for the data set currently on that volume.

SYSTEM

This field represents the NODEID value for the system that currently owns the volume. This field is blank for volumes in scratch status.

DEST

This field indicates where synchronization information is sent for this volume. SYS1 indicates that information is sent to the system represented by the data sets SYS1IN and SYS1OUT and displayed in the first column of the report. SYS2 represents the other system. BOTH indicates that both systems are to receive information about this volume. NONE indicates that a critical data error was detected for this volume. The MESSAGE field contains more information.

MESSAGE

This field contains additional information for determining why a discrepancy occurred. If this field is blank, a discrepancy was found in the information stored for the volume (data set name, creation date, and so on). The synchronization utility assumes that the volume with the most current creation date and time is accurate and sends this information to the other system. See the *Messages Reference Guide* for a detailed explanation of the messages.

TOTAL RECORDS READ

This field displays the total number of records (excluding the identification record) that were read from the SYSxIN files.

TOTAL RECORDS WRITTEN

This field displays the total number of records (excluding the identification record) that were written to the SYSxOUT files.

TOTAL RECORDS INVALID

This field displays the total number of volumes that contained invalid information. These volumes have the value NONE in the DEST field of the report.

DYNUPDTE Utility—Update the CA Dynam/T Databases

Use the DYNUPDTE utility to attempt to resolve the discrepancies found by the synchronization utility.

DYNUPDTE uses the synchronization information from DYNSYNC.

CA 1 must be active or batch active to run DYNUPDTE.

Report Description

DYNUPDTE produces two reports: TMS Report 71 - CA 1 CA Dynam/T Interface Update Report and TMS Report 72 - CA 1 CA Dynam/T Interface Exception Report.

The Interface Update Report contains detailed information about all updates and errors that occurred during processing. A sample of this report appears on page TMS Report 71 - CA 1 CA Dynam/T Interface Update Report (see page 80).

The Interface Exception Report contains only the exceptions; the information in this report requires further investigation and manual database correction. A sample of this report appears on page TMS Report 72 - CA 1 CA Dynam/T Interface Exception Report (see page 80).

Job Control Statements

```
//stepname EXEC PGM=DYNUPDTE[,PARM='[DATEFMT=(fmt)][,TEST]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TMSRPT2  DD  SYSOUT=A
//UPDATES  DD  DSN=cai.ca1.syncdata,DISP=SHR
```

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Reports 71 and 72. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information about valid date formats, see Overriding the Preferred Date Pattern (see page 21).

TEST

Indicates that no updates are made.This parameter is optional.

JCL Considerations

UPDATES DD

This statement points to the data set created by DYNSYNC for CA 1.

TMS Report 71 - CA 1 CA Dynam/T Interface Update Report

TMS Report 71 is produced by specifying the TMSRPT DD statement in the DYNUPDTE JCL.

```
DATES=04/01/95.091                               Y O U R   C O M P A N Y   N A M E                                  PAGE=00001

TMS REPORT-71  CA 1 CA DYNAM/T INTERFACE UPDATE REPORT

VOLSER  DATA SET NAME   STAT SYSTEM       CREATED      EXPDT      ACTION    MESSAGE

000003 DYNT.DATASET.D003 ACT DYNAMT   MMMDD YYYY 1200 PERM       UPDATED
000007 DYNT.DATASET.D007 ACT DYNAMT   MMMDD YYYY 1200            REJECTED ERR02: CANNOT UPDATE CA 1 OWNED VOLUME
000009 DYNT.DATASET.D009 SCR          MMMDD YYYY 1200 MMMDD YYYY UPDATED
000010 CA01.DATASET.D010 SCR CA1MVS   MMMDD YYYY 1200 MMMDD YYYY UPDATED
000013 DYNT.DATASET.D013 SCR          MMMDD YYYY 1200            REJECTED ERR06: VOLUME IN DELETE STATUS ON TMC
000014 CA01.DATASET.D014 ACT CA1MVS   MMMDD YYYY 1200            REJECTED ERR06: VOLUME IN DELETE STATUS ON TMC
000015 DYNT.DATASET.D015 ACT DYNAMT   MMMDD YYYY 1200            REJECTED ERR06: VOLUME IN DELETE STATUS ON TMC
000101                   SCR                                     REJECTED ERR01: VOLUME NOT IN TMC
000102                   SCR                                     REJECTED ERR01: VOLUME NOT IN TMC
000103                   SCR                                     REJECTED ERR01: VOLUME NOT IN TMC
000104                   SCR                                     REJECTED ERR01: VOLUME NOT IN TMC
000105                   SCR                                     REJECTED ERR01: VOLUME NOT IN TMC

        12 TOTAL EXTRACT RECORDS
         3 TOTAL VOLUMES UPDATED
         9 TOTAL EXTRACTS REJECTED
```

TMS Report 72 - CA 1 CA Dynam/T Interface Exception Report

TMS Report 72 is produced by specifying the TMSRPT2 DD statement in the DYNUPDTE JCL.

```
DATES=04/01/95.091                               Y O U R   C O M P A N Y   N A M E                                  PAGE=00001

TMS REPORT-72  CA 1 CA DYNAM/T INTERFACE EXCEPTION REPORT

    ------- CA DYNAM/T TO CA 1 EXTRACT ------- -------  TMC VOLUME REC  ORD  -------
VOLSER   DATA SET NAME  STAT SYSTEM       CREATED       DATA SET NAME  STAT     CREATED           MESSAGE

000007 DYNT.DATASET.D007 ACT DYNAMT MMMDD YYYY 1200 CA01.DATASET.D007 ACT MMMDD YYYY 1200 ERR02:CANNOT UPDATE CA 1 OWNED VOLUME
000013 DYNT.DATASET.D013 SCR        DMMMDD YYYY1200                                       ERR06:VOLUME IN DELETE STATUS ON TMC
000014 CA01.DATASET.D014 ACT CA1MVS MMMDD YYYY 1200                                       ERR06:VOLUME IN DELETE STATUS ON TMC
000015 DYNT.DATASET.D015 ACT DYNAMT MMMDD YYYY 1200                                       ERR06:VOLUME IN DELETE STATUS ON TMC
000101                   SCR                                                              ERR01:VOLUME NOT IN TMC
000102                   SCR                                                              ERR01:VOLUME NOT IN TMC
000103                   SCR                                                              ERR01:VOLUME NOT IN TMC
000104                   SCR                                                              ERR01:VOLUME NOT IN TMC
000105                   SCR                                                              ERR01:VOLUME NOT IN TMC

       12 TOTAL EXTRACT RECORDS
        3 TOTAL VOLUMES UPDATED
        9 TOTAL EXTRACTS REJECTED
```

DYNUPDTE Report Field Definitions

VOLSER

This field contains the volume serial number associated with the transaction.

DATA SET NAME

This field contains the last 17 characters of the data set name to be placed in the Volume record.

STAT

This field indicates that the status of the volume was changed. SCR indicates that the volume is being placed into scratch status. ACT indicates that the volume is being placed into NONSCRATCH status.

SYSTEM

This field indicates the NODEID of the system that owns the volume. This field may be blank if the volume is to be scratched.

CREATED

This field contains the creation date and time associated with the data set on the volume.

EXPDT

This field indicates what the expiration date is after the update is performed.

ACTION

This field indicates whether the transaction was successfully applied to the TMC. UPDATED means the update was successful. REJECTED indicates that an error was encountered. See MESSAGE for more details.

MESSAGE

This field contains the message associated with transactions that have been rejected. Possible messages are listed in the *Messages Reference Guide*.

TOTAL EXTRACT RECORDS

Number of Volume record updates attempted.

TOTAL VOLUMES UPDATED

Number of Volume record updates processed successfully.

TOTAL EXTRACTS REJECTED

Number of Volume record updates rejected.

L0C6INIT Utility—Initialize and Reinitialize CA 1

Use the L0C6INIT utility to initialize and reinitialize CA 1.

L0C6INIT uses the facilities from CAIRIM. It is responsible for installing the CA 1 SVC, building the TMS subsystem, initializing the WTO subsystem interface, loading realtime modules to common storage if not found in the link pack area, installing the CA 1 OSI to track tape activity, locating and anchoring CA 1 realtime user exits, and installing the SMF interface to capture tape read and write errors.

L0C6INIT is a required phase of the CA 1 initialization process. At system startup time, the CAIRIM procedure invokes the L0C6INIT module when the control statements are used as input to the PARMLIB DD of CAIRIM.

Important! This program should be used only by systems programmers responsible for maintaining CA 1.

CA 1 can be active, batch active, or inactive to run L0C6INIT.

Note: If you have other vendor products which modify the operating system's data management modules, and these modified versions MLPAd, you must include the CAG8LIB2 DD in the CAIRIM procedure and the TMSINIT procedure. Any MLPA data set must be specified in the CAG8LIB2 DD which contains these modifications. If you have a data set name for the contents of the LPA other than SYS1.LPALIB, you must put that data set name in the CAG8LIB1 DD and include the DD in the CAIRIM procedure as well as the TMSINIT procedure. Ensure that both the CAIRIM procedure (CAS9) and the TMSINIT procedure have security access (READ) to the libraries containing the operating system's data management modules. If all of the open/close routines are not copied to the modified library, then you must have SYS1.LPALIB in the concatenation with CAG8LIB1. Update authorization is required for the following resources in the FACILITY class:

- CSVDYLPA.ADD.IFG019CA
- CSVDYLPA.DELETE.IFG019CA

Report Description

L0C6INIT does not generate any reports.

Job Control Statements

The following control statement is specified in the PARMLIB DD of the CAIRIM procedure when L0C6INIT services are required:

```
                                                                       72
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)          -
PARM(INIT,SVC=xxx [,OSI=YES] [,LPA=] [,SMF=YES])
```

This control statement directs L0C6INIT to initialize CA 1. The SVC=xxx keyword specifies the user SVC number, in the range of 200 to 255, to be assigned to the CA 1 SVC TMS00SVC. TMS00SVC is installed as the SVC specified, unless it has been installed during system startup time using the SYS1.PARMLIB IEASVCxx member. If the module is not found in the link pack area, L0C6INIT loads it in extended common storage, using CAIRIM dynamic SVC services, and updates the SVC table. The INIT function can be performed only once. Subsequent invocations of L0C6INIT are required to specify the REINIT parameter.

Note: For more information about the CA 1 OSI, see the *Programming Guide*. For more information about batch activating or deactivating CA 1, see TMSINIT (see page 328) in this guide.

Once the initial L0C6INIT has run, the REINIT parameter can be used to reinitialize certain components of CA 1. The following control statement is specified in the PARMLIB DD of the CAIRIM procedure:

```
                                                                       72
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)          -
PARM(REINIT [,SVC=xxx] [,LPA=xxxxxxxx] [,WTO=xxxxx] [,OSI=xxxxx]        -
[,SMF=xxx] [,COSI=xxx]) [LOADLIB(authorized.library)]
```

The REINIT parameter will reinitialize the components specified by the other keywords provided in the CAIRIM PARM control statement. Optionally, the LOADLIB parameter can be loaded from a specific authorized load library instead of from the link list concatenation.

Control Statement Definition

INIT

Directs L0C6INIT to perform CA 1 Initialization. Can be used only once per IPL.

REINIT

The REINIT keyword is used on subsequent L0C6INIT requests to reinitialize or refresh certain components of CA 1.

LPA=

The LPA= statement directs L0C6INIT to load the specified resident module into common storage, or all resident modules if ALL is specified.

ALL

Specifies that all CA 1 commonly accessed, resident routines are to be reloaded, and CA 1 pointers refreshed.

modname

Specifies the name of the module to be added to common storage.

WTO=

This control statement directs L0C6INIT to either temporarily or permanently shut down the CA 1 WTO and DOM subsystem interface (WTO SSI). The WTO SSI is used to capture and reissue tape mount messages.

NO

Performs just like WTO=FORCE. This conforms more to other options and shortens up the PARMS input.

CLEAR

Temporarily disables the WTO SSI. A subsequent L0C6INIT REINIT request enables the WTO SSI again.

FORCE

Shuts down the WTO SSI for the remainder of the life of the IPL. The WTO SSI is not reactivated until the system is IPLed again.

SMF=YES

This specifies that L0C6INIT should attempt to load and install any SMF exit modules.

OSI=

This control statement directs L0C6INIT to install a new version of the G8/API table, not load OSI, turn on proper trace options, or turn off trace options previously set.

DEBUG

Turn on proper trace options.

NODEBUG

Turn off trace options previously set.

REINIT

Only use this parameter to install a new version of the G8/API table.

COSI=YES

Reinitialize catalog interface.

DEBUG

Turn on the catalog debugging option.

NODEBUG

Turn off the catalog debugging option.

NO

Do not install the catalog interface. If it is already installed, remove the interface.

SYS={00|xx}

Defines the initialization members for CA 1 where xx is a user-defined two-character alphanumeric or national (@,#,$) value. The default is 00 and refers to TMOSYS00. When using xx for initialization, TMOSYS00 must also be present.

TMSINIT loads the CA 1 SMF Accounting exits. Some clients have to enable the accounting exits early in the IPL process. You can now have L0C6INIT execute TMSINIT directly. Include the //TMSOPTNS, //TMSRPT, //TEMPAUDT, and //TMSPARM DD statements from TMSINIT into the CAS9 proc itself.

If you are using the Real-time Retention Data Set (RDS), also include the //TMSRDS DD statement.

This parameter is strictly optional. The parameter enables you to execute TMSINIT as part of the CAS9 process itself. The functionality is not executed in a stand-alone fashion after CAS9 has completed.

If using L0C6INIT to execute TMSINIT directly, the START TMSINIT in the CAUTOCMD or COMMNDxx member in SYS1.PARMLIB is not required.

More information:

Job Control Statements (see page 334)

Control Statement Examples

For the initial execution of L0C6INIT during system startup time, use the following control statements:

```
                                                                      72
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)         -
PARM(INIT,SVC=xxx)
```

In addition to INIT and SVC=xxx, you can specify the LPA=, SMF=, and the WTO= keywords.

To load a new version of the CA 1 realtime user exit TMSXITC:

```
                                                                      72
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)         -
PARM(REINIT,LPA=TMSXITC)                                               -
[LOADLIB(authorized.library)]
```

The LOADLIB parameter can be specified if the new version of TMSXITC is not found in the current link list concatenation. The REINIT causes the stored address of TMSXITC in the CA 1 resident table to be refreshed. If additional modules are to be reloaded, the three statements above may be repeated with a different module name specified on the LPA= parameter.

The SVC= keyword can be used on a reinitialization request (REINIT), but is not normally required. SVC=xxx could reassign TMS00SVC to a different user SVC number if the original specification overlaid another user SVC in the SVC table.

To temporarily shutdown the WTO SSI *only*:

```
                                                                      72
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)         -
PARM(WTO=CLEAR)
```

To reactivate the WTO SSI:

```
                                                                      72
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)         -
PARM(REINIT)
```

The REINIT will refresh pointers in the CA 1 resident table. Resident routines and user exits are relocated and the SMF exits are reinitialized.

TMSAGGR Utility—Maintain Data Set Name Blocks

Use the TMSAGGR utility to maintain Data Set Name Blocks (DSNBs) containing information on secondary files that are associated with the Volume record.

TMSAGGR cannot be used for tapes controlled by an External Data Manager (EDM). The program builds DSNBs to facilitate the initial creation of a TMC to store information on secondary data sets. When DSNBs are found to be in error, the utility can be executed to correct TMC records that have invalid DSNB data associated with them, or to generate new DSNBs to be associated with a Volume record to accommodate off-site processing. The utility TMSUDSNB can be used to correct fields in DSNB records.

CA 1 does not require DSNBs to be associated with a volume to read secondary files for input. However, DSNBs are required if you want to add additional files to a volume or append data (DISP=MOD) to the last file currently on a volume.

It is not necessary to purge DSNBs from a volume that you want to expire and go scratch; if File 1 is manually expired, the TMSCLEAN utility allows the volume to go scratch even if DSNBs with a higher expiration date are present on the volume.

CA 1 must be active or batch active to run TMSAGGR.

Report Description

TMSAGGR produces TMS Report 20 - Addition and Deletion of Data Sets. This report includes a listing of all processed control statements and provides totals for DSNBs added, deleted, allocated and available. If any errors are encountered during processing, TMSAGGR sets the completion code to 12. Errors are flagged on the report output. A sample of this report appears on page TMS Report 20 - Addition and Deletion of Data Sets (see page 93).

Job Control Statements

```
//stepname EXEC PGM=TMSAGGR
//             [,PARM='[DATEFMT=(fmt)][,SCAN=xx][,DEFAULT][,TEST]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//SYSIN    DD  *
(control statements here)
/*
```

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 20. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

SCAN

Specifies the number of positions to scan in the control statements for use in TMSAGGR.

SCAN={72|68|xx}

Specifies that xx is the continuation column with a maximum value of 72. If TMSIDATA is used to create the control statements, PARM='SCAN=68' is required. The default is 72.

DEFAULT

Indicates that all new records are to have the eligible for RDS override FLAG1 bit set to allow for processing by the Retention Data Set.This parameter is optional.

TEST

Indicates that no updates are made to the TMC. This parameter is optional.

JCL Considerations

SYSIN

Defines the control statement data set. This can be any sequentially readable data set.

Control Statement Specification

Three control statements are used with TMSAGGR: the VOL control statement, the DSNB data control statement and the DEL control statement. *The first control statement must be DEL or VOL*.

You can also include comment statements.

VOL Control Statement

The VOL control statement identifies the TMC record to be updated. It is *required* when generating or purging DSNBs. If DSNBs are being generated, the VOL control statement is followed by DSNB data control statements. If DSNBs are being released, no DSNB data control statements are allowed between the VOL control statement and the next VOL or DEL control statement. The format for the VOL control statement is:

```
1   5
VOL nnnnnn[,PURGE]
```

Control Statement Definition

VOL

Is the control statement identifier. VOL must start in position 1, and must be followed by one space.

nnnnnn

Is the volume serial number of the TMC record to be updated. This parameter can be five or six characters in length.

PURGE

Is used to indicate that all DSNBs associated with the specified volume serial number are to be purged.

DSNB Data Control Statement

DSNB data control statements generate DSNBs for secondary files on a volume. No DSNBs can currently be associated with the volume. For example, if you need to provide DSNB information for File 5, and the Volume record currently references the DSNBs associated with files 2 through 4, these DSNBs must be purged with TMSAGGR and the entire chain rebuilt.

DSNB data control statements *must* follow the VOL control statement and *must* be in file sequence order beginning with the DSNB data for File 2. For multivolume, multi-data set tape aggregates, all associated DSNB data control statements must follow one single VOL control statement. Keyword F1STVSN may be used to define the structure of the aggregate if not all data sets begin on the same volume. The DSNBs must be chained to the volume specified on the VOL control statement or to subsequent volumes of the chain. If F1STVSN is not used, all generated DSNBs are chained to the volume from the VOL control statement.

The requirements and restrictions for a DSNB data control statement are:

- A DSNB data control statement can consist of several keywords. The first keyword should begin in position 1.
- Continuation of a DSNB data control statement is indicated by placing a nonblank character in position 72, or if the SCAN=xx parameter was specified, xx indicates the position for a continuation character. Keywords placed on continued lines should begin in position 16 and not exceed position 71, or xx-1.
- A keyword and its value (keyword=value) must not span two control statements.
- All data entered to construct a DSNB must be in one control statement. Continuations must be used until all required data for a DSNB is present in the control statement.
- Each DSNB data control statement must contain a data set name and an expiration date.
- Each DSNB keyword can be used only once per DSNB data control statement.
- HEXZEROS is not supported as a symbolic value to direct that a field be set to hexadecimal zeros.

The format for the DSNB data control statement is:

```
1              16                                                      72

keyword=value,keyword=value                                            *
               keyword=value
```

Control Statement Definition

The following keywords can be specified:

DSN or D

Is the data set name to be associated with the DSNB. Each DSNB data control statement *must* contain the DSN keyword. Data set names that contain special characters must be properly delimited. This keyword is required for each DSNB data control statement.

For example, a data set name of BACKUP.#WKLY must be enclosed by characters whose EBCDIC numeric value is less than the numeric value of the character A (+,!,*,%,?,#,@,",=), such as DSN='BACKUP.#WKLY' or DSN=*BACKUP.#WKLY*

EXPDT or E

Specifies the data set expiration date. Value is a 10-character field in preferred date format or Julian date format (yyyy/ddd). Expiration date types should not be mixed on a tape volume. CA 1 expiration date keywords should not be mixed with Julian dates or other CA 1 expiration date keywords. If more than one type of expiration date is found in a volume, a warning message is issued if MIXEXP is set to WARNING (the default) in the TMOOPTxx member of hlq.CTAPOPTN. This keyword is *required* for each DSNB data control statement.

Important! CA 1 controls the expiration of tapes based on the EXPDT for File 1. When adding DSNBs to a volume, keep in mind that if a numerically higher EXPDT value is used for a DSNB, the Volume record for File 1 is updated with this higher value.

CJOB or J

Is used to enter the creating job name into the DSNB. If omitted, the job name for the TMSAGGR execution is used.

CSTEP

Is used to enter the creating step name into the DSNB. If omitted, TMSAGGR is used as the default value.

CDATE or C

Specifies the date of data set creation in preferred date format or Julian date format. If omitted, the date of the current execution is used as the default value.

CPGM

Used to enter the creating program into the DSNB. If omitted, the field is left blank.

CTIME

Is used to enter the data set creation time into the DSNB. This field is a four-position numeric field in the form hhmm. If omitted, the current execution time is used as the default value.

BLKSIZE

Specifies the data set block size in the DSNB. This field, if used, must contain from one to five digits. If omitted, zero is assigned.

BLKCNT

Specifies the number of record blocks in the data set. Value must be a numeric field from one to six digits. If omitted, zero is the default.

LRECL

Specifies the data set logical record length in the DSNB. If this keyword is used, value must be a numeric field from one to five digits. The default value is zero.

RECFM

Specifies the data set record format in the DSNB. Value is a field which contains two hexadecimal characters. The default value is 00.

F1STVSN

Can be used for multivolume, multi-data set aggregates and specifies the volume where the data set begins. Value must be a valid VOLSER out of the multivolume chain. The F1STVSN values of all DSNBs must be in volume sequence order, beginning with the volume specified on the VOL control statement or a subsequent volume of the chain.

If omitted, the default value is the F1STVSN value of the previous DSNB. For the first DSNB, the default value is the volume from the VOL control statement.

The following example shows how the DSNBs for files 2, 3 and 4 of volume 100000 are allocated to the Volume record.

```
1              16                                                      72

VOL 100000
DSN=TMS.PRODOTT2,EXPDT=DDMMMYYYY,C=DDMMMYYYY,                          *
               CJOB=ABCGTMC
DSN=TMS.PRODOTT3,EXPDT=DDMMMYYYY,C=DDMMMYYYY,                          *
               CJOB=ABCGTMC
DSN=TMS.PRODOTT4,EXPDT=DDMMMYYYY,C=DDMMMYYYY,                          *
               CJOB=ABCGTMC
```

DEL Control Statement

The DEL control statement is used to return to the free chain of DSNB records, a used DSNB that does not have a valid association with a Volume record (the DSNB active indicator is set when it is used.) The TMSPTRS program, which analyzes errors in the TMC multivolume, multi-data set and DSNB free chains identifies DSNBs to be deleted by TMSAGGR.

The format for the DEL statement is:

```
1   5

DEL nnnnnnn
```

Control Statement Definition

DEL

Is the control statement identifier. It must start in position 1 and be followed by one space.

nnnnnnn

Is the one- to seven-digit number of the DSNB, which is being returned to the DSNB free queue. The ACTIND of the DSNB to be deleted must be equal to X'80'.

Comment Statement

You can include comment statements by placing an asterisk (*) in position 1. Text for a comment can appear anywhere between positions 2 and 80 inclusive.

Control Statement Examples

1. Assume that TMSPTRS identified several used DSNBs (numbers 000315, 000050 and 001388) that no longer had valid associations with a volume. The TMSAGGR control statements to clear the DSNBs of information and return them to the free chain are:

```
DEL 000315
DEL 000050
DEL 001388
```

2. VOLSER 109830 was taken off-site and five files were written on it. The control statements to build the DSNB chain are:

```
                                                                      72

VOL 109830
DSN=FILE.TWO.DATASET.NAME,EXPDT=*MMDD YYYY*,CJOB=OFFSITE,              *
               CDATE=*MMMDD YYYY*,CTIME=1530
DSN=FILE.THREE.DATASET.NAME,EXPDT=*MMMDD YYYY*,CJOB=OFFSITE,           *
               CDATE=*MMMDD YYYY*,CTIME=1533
DSN=FILE.FOUR.DATASET.NAME,EXPDT=*MMMDD YYYY*,CJOB=OFFSITE,            *
               CDATE=*MMMDD YYYY*,CTIME=1539
DSN=FILE.FIVE.DATASET.NAME,EXPDT=*MMMDD YYYY*,CJOB=OFFSITE,            *
               CDATE=*MMMDD YYYY*,CTIME=1547
```

3. The control statements to purge the DSNBs from volumes 153100, 158309 and 143772 are:

```
VOL 153100,PURGE
VOL 158309,PURGE
VOL 143772,PURGE
```

TMS Report 20 - Addition and Deletion of Data Sets

The following example shows the creation of DSNB records for volumes 000000, 000073 and 000072.

```
TUESDAY, MARCH 18, 2005.199                      Y O U R   C O M P A N Y   N A M E                             ROxxxxx PAGE=00001

TMS REPORT 20 -      - ADDITION AND DELETION OF DATA SETS


VOL 000000
DSN=ADMIN.TEST.BURMA.F2,CJOB=ADMINMB,CDATE=95177,CTIME=0825,E=95182
DSN=ADMIN.TEST.BURMA.F3,CJOB=ADMINMB,CDATE=95177,CTIME=0825,E=95182
DSN=ADMIN.TEST.BURMA.F4,CJOB=ADMINMB,CDATE=95177,CTIME=0825,E=95182
DSN=ADMIN.TEST.BURMA.F5,CJOB=ADMINMB,CDATE=95177,CTIME=0825,E=95182

VOL 000073
DSN=SYSPROG.TEST.BLOKNOR.S02,CJOB=SYSPRJK,E=LDATE/014
DSN=SYSPROG.TEST.BLOKNOR.S03,CJOB=SYSPRJK,E=LDATE/014
DSN=SYSPROG.TEST.BLOKNOR.S04,CJOB=SYSPRJK,E=LDATE/014
DSN=SYSPROG.TEST.BLOKNOR.S05,CJOB=SYSPRJK,E=LDATE/014
DSN=SYSPROG.TEST.BLOKNOR.S06,CJOB=SYSPRJK,E=LDATE/014
DSN=SYSPROG.TEST.BLOKNOR.S07,CJOB=SYSPRJK,E=LDATE/014

VOL 000072
DSN=SYSPROG.TEST.BLOKNOR.SL2,CJOB=SYSPRJK,E=LDATE/014
DSN=SYSPROG.TEST.BLOKNOR.SL3,CJOB=SYSPRJK,E=LDATE/014

          12 DSNBS ADDED             0 DSNBS DELETED       12,371 DSNBS USED        12,629 DSNBS AVAILABLE
```

In the example below, six DSNBs were deleted and returned to the free queue: four were purged from Volume record 000871 and two used DSNBs that were not properly attached to a Volume record were deleted.

```
SUNDAY, MARCH 25, 2005.176                       Y O U R   C O M P A N Y   N A M E                             ROxxxxx PAGE=00001

TMS REPORT 20 -      - ADDITION AND DELETION OF DATA SETS


VOL 000871,PURGE

DEL 104                                                                                    DSNB DELETED

DEL 1023                                                                                   DSNB DELETED

           0 DSNBS ADDED             6 DSNBS DELETED       12,365 DSNBS ALLOCATED       12,635 DSNBS AVAILABLE
```

TMSAGGR Report Field Definitions

Control statement image

Input control statement or parameter.

Message

Processing messages for DSNB records.

DSNBS ADDED

Number of DSNB records processed by ADD command.

DSNBS DELETED

Number of DSNB records processed by DELETE command and PURGE keyword.

DSNBS USED

The total number of used DSNBS from TMSCTL#2.

DSNBS AVAILABLE

Number of unused DSNB records after processing.

TMSAUDEX Utility—Extract TMC Audit Data

Use the TMSAUDEX subprogram of the TMSXTEND utility to build a table of systems sharing the TMC.

The table is constructed by extracting records from the CA 1 Audit data set and is then used by TMSXTEND to manage the switch-over to a newly constructed TMC with new volume ranges or DSNBs. TMSAUDEX also checks to ensure that all systems sharing the TMC are at the proper level necessary to permit a TMC extend function.

TMSAUDEX is also being documented separately because it can be executed as a standalone utility to examine the active Audit data set or prior generations of Audit data sets. The utility prints out the number of Audit records for each system sharing the TMC and the start and end date/time of the Audit records in the file.

Report Description

TMSAUDEX creates TMS REPORT-81 which displays all z/OS systems sharing the TMC as recorded in the Audit file. The first and last dates of the Audit records for each system are also displayed and an indicator showing that CA 1 is at the necessary level to perform an extend function.

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

Job Control Statements

```
//JOBAUDEX JOB (ACCOUNT)
//**------------------------------------------------------------------*
//**                                                                   *
//**       TMSAUDEX INSPECTS THE CA 1 AUDIT FILE TO                    *
//**       DETERMINE WHICH CPUS ARE SHARING THE TMC.                   *
//**                                                                   *
//**------------------------------------------------------------------*
//TMSAUDEX PROC LINKLIB='CAI.CTAPLINK',
//             AUDIT='YOUR.AUDIT.FILE',
//             SOUT='*'
//TMSAUDEX EXEC PGM=TMSAUDEX,REGION=0M
//STEPLIB    DD DSN=&LINKLIB.,
//             DISP=SHR
//AUDIT      DD DSN=&AUDIT.,
//             DISP=SHR
//TMSRPT81   DD SYSOUT=&SOUT.
//           PEND
//AUDEX    EXEC TMSAUDEX
//TMSAUDEX.CPULIST DD *
(CONTROL STATEMENTS HERE)
/*
//
```

Parameter Definitions

There are no JCL execution parameters for TMSAUDEX.

JCL Considerations

AUDIT

Defines the CA 1 Audit data set to be processed. You may also specify previous versions of the Audit data set on either disk or tape. If this DD statement is omitted, the active Audit data set will be processed.

CPULIST

This DD defines the control statement input data set.

TMSRPT81

Defines the report output data set for the TMS REPORT-81. This DD statement is required.

Completion Codes

TMSAUDEX adheres to the standard CA 1 conventions for batch utilities, issuing user abends for problems common to most CA 1 utilities.

0

Program completed successfully.

8

One or more systems sharing the TMC is not at the maintenance level required by the TMSXTEND utility.

U0001

CA 1 is not active or the AUDIT DD statement does not point to an Audit data set.

U0002

TMSRPT81 DD statement is missing from the JCL.

U0006

Invalid data. This can be issued for invalid or incomplete control statements specified in the CPULIST DD as well as for invalid data found in the Audit file. In environments where the TMC is shared with VM:Tape a U0006 is generated if VM:Tape does not have the required level of maintenance applied. A blank CPU ID in the TMS REPORT-81 is indicative of VM:Tape being back level. With the required maintenance applied, VM:Tape will display a CPU ID that can be customized by the user and defined by VM:Tape.

Control Statement Specifications

The control statements for TMSAUDEX are designed for use when the program is run as part of a TMSXTEND function. When run in standalone mode TMSAUDEX will always attempt to report on all systems found in the Audit file. If specified, the INCLUDE and EXCLUDE control statements are only verified for proper syntax.

Following is a complete discussion on using the INCLUDE and EXCLUDE statements associated with running the TMSXTEND utility.

EXCLUDE smf id

This control statement defines the 4-byte SMF id of systems running CA 1 that should be excluded from processing by TMSXTEND. EXCLUDE must begin in column 1 followed by a blank and then the SMF id. Only one SMF id may be specified per control statement. Note that failure to find a system specified on an EXCLUDE statement is not considered an error. TMSAUDEX will continue to process and complete with a RC=0. You should never exclude a system that is active and using TMC that you are extending.

INCLUDE smf id

This control statement defines the 4-byte SMF id of a system running CA 1 that should be included in the list of systems to process when performing an extend function. INCLUDE must begin in column 1 followed by a blank and then the SMF id. Only one SMF id may be specified per control statement.

Note: This utility supports comment statements that begin with an asterisk (*) in column 1; however, they are not listed in any report. Comments may also be placed on the INCLUDE and EXCLUDE statements after column 21.

TMS Report 81 - Report Field Definitions

```
THURSDAY, MAY 02, 2013.122                Y O U R   C O M P A N Y   N A M E
P02138 PAGE=00001
TMS REPORT-81      AUDIT EXTRACT BY CPU

  CPU  RECORD COUNT  LO--DATE    TIME   HI--DATE    TIME FLG URC URA CTL STMT ERRORS
  CPU1        1,815  16APR,2013  13:17  01MAY,2013  16:32 Y   N   N
  CPU2          470  19APR,2013   0:44  19APR,2013  11:07 Y   N   N
 ************************************
  * CONTROL STATEMENTS PROCESSED:     *
 ************************************
 NO CONTROL STATEMENTS PROCESSED
```

CPU

The SMF id of the z/OS system creating audit records. SMF ids for systems that are specified on an INCLUDE or EXCLUDE statement are also listed even if no audit records were found for the requested system. Users running VM Tape and sharing the TMC with that product can see a report line with a blank SMF id. The blank SMF id is an indication of the VM Tape system not being at the necessary maintenance level for sharing the TMC and is considered an error.

RECORD COUNT

Total number of audit records that are found for the system.

LO--DATE/TIME

Lowest date and time of any audit record that is found for the system.

HI--DATE/TIME

Highest date and time of any audit record that is found for the system.

FLG

Flag indicating the system is at the required level to run the TMC extend utility TMSXTEND. All listed systems must be at the required level to run TMSXTEND.

URC

(Unique_RNAME Capable) Flag indicating the system is at the required level to utilize the Unique_RNAME capability.

URA

(Unique_RNAME Active) Flag indicating the system currently has the Unique_RNAME capability active. For more information about the Unique_RNAME capability, see the *Programming Guide.*

CTL STMT

This column shows INCLUDE or EXCLUDE if a control statement was specified for this system.

ERRORS

This column provides a description of any errors that are encountered in processing the Audit file.

Errors - ERROR WRONG CODE LEVEL

Reason:

This message will be issued by TMSAUDEX if TMSINIT has not run on that system or if there has been no tape processing since the last backup or if the system does not have the required maintenance to support TMSXTEND. If the system has the required maintenance to support TMSXTEND, then run TMSINIT or a job to create a tape file via a program not linked as an authorized program, such as CTSTAPER, and then rerun the procedure. If the system does not have the required maintenance to support TMSXTEND, then apply the correct maintenance to support TMSXTEND, run TMSINIT, and then rerun the procedure.

Action:

Run TMSINIT or a job to create a secondary file and rerun procedure.

TMSAUDIT Utility—Analyze TMC Activity

Use the TMSAUDIT utility to analyze activity that has occurred within the TMC.

A Volume record, several specific Volume records or a range of Volume records may be targeted. This can be useful for audit, accounting purposes and problem analysis.

Audit records can be listed in three different formats: SHORT, LONG or DUMP. (The LONG format provides descriptions of all of the bit settings for the flag bytes.) Control Record reports are automatically generated in both character and hexadecimal formats. Audit records from the Audit data set or from the Audit backup tapes provided by TMSCOPY can be used as input to produce up to seven possible reports. See the *Administration Guide*. Sample Weekly Batch Flowchart and Sample Monthly Batch Flowchart for examples of concatenating Audit data set backups for reporting purposes.

CA 1 must be active or batch active to run TMSAUDIT.

Report Description

TMSAUDIT generates TMS Report 32:

- Audit Data Set Activity
- 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

When requesting information in SHORT format the fields printed on the report for Audit records (0 through 4) are: volume serial number or DSNB number, volume sequence, file sequence, data set name, expiration date, FLAG1, FLAG2, FLAG3, FLAG4 bits and time stamp area information.

The first page of TMS Report 32 lists the keys to interpret FLAG1, FLAG2, FLAG3 and FLAG4. For each flag, the hexadecimal values can be added to denote combinations of events. For example, X'41' for FLAG1 indicates CA 1 tracked the CLOSE of the volume and the volume is eligible for RDS override, X'40' + X'01' = X'41'.

The listing on internal processing records (Type 5 Audit records) contains control statement or message input, the date and time the record was written, and the name of the function and the CPU ID which wrote the record.

Samples of these reports are provided, beginning on TMS Report 32 - Audit Data Set Activity (see page 109).

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

Job Control Statements

```
//stepname EXEC PGM=TMSAUDIT[,PARM='DATEFMT=(fmt)']
 //STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
 //TMSRPT   DD  SYSOUT=A
[//AUDIT    DD  DSN=audit.backup.tape,DISP=OLD]           *NOTE
 //SYSOUT   DD  SYSOUT=A
 //SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
 //SORTWK01 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SORTWK02 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SORTWK03 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SORTWK04 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SORTWK05 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SORTWK06 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SYSIN    DD  *
  (control statements here)
 /*
```

Note: The Audit DD is optional and may be an existing backup tape of the Audit data set. If Audit DD is not present, the current Audit disk production data set is used.

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 32. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

Control Statement Specification

TMSAUDIT accepts VOLSER, TYPE, SORT, DATE, TIME, DSN and FORMAT control statements. The TYPE control statement is *required*. Each control statement can be entered only once for each program execution.

VOLSER Control Statement

The VOLSER control statement is optional. It is used to select up to 10 individual volume serial numbers or a range of volume serial numbers to be printed. The format for the VOLSER control statement is:

```
1      8

VOLSER=vvvvvv[,...,vvvvvv]
  or
VOLSER=vvvvvv-vvvvvv
```

Control Statement Definition

VOLSER

Is the control statement identifier and must start in position 1.

vvvvvv

Indicates the volume serial number(s) of the tape(s) to be reported. When two volume serial numbers are separated by a dash (-), all volume serial numbers in this inclusive range are selected and printed on the Audit report. When two or more volume serial numbers are separated by commas, each VOLSER is reported on individually. Up to 10 volume serial numbers can be specified.

Note: VOLSERs less than 6 characters in length must be padded to the right with blanks. If a range is specified, the dash (-) must be in column 14.

TYPE Control Statement

The TYPE control statement specifies the types of audit records to print. This control statement is *required*. The format for the TYPE control statement is:

```
1    6

TYPE=n[,...,n]
```

Control Statement Definition

TYPE

Is the control statement identifier and must start in position 1.

n

Is the type of Audit record. A maximum of six types can be requested. The allowable values for this type are:

Type

Definition

0

TMC records which have been updated manually or through CA 1 batch programs.

1

Tapes which have been opened/closed for input.

2

Tapes which have been opened/closed for output.

3

Exception records. These records are flagged with exception codes. The exception codes and their meaning are explained in the following section:

002— Image before updates if the AUDB4 option is set to ALL or BATCH.

003— Nonresident multifile record for output.

004— Nonresident input volume.

005— Nonresident output volume.

006— Nonresident multivolume.

007— Nonresident tape initialized.

008— Control Record manually updated.

013— A mixed expiration date was encountered and the system set the option MIXEXP to WARNING or LOG.

014— A mixed expiration date was encountered and the system set the option MIXEXP to DISALLOW.

CLA— TMSSMF83 creates this record. The operating system cataloged this data set, which was already indicated as being cataloged within the TMC. The bit TMISCAT of TMFLAG4 for the volume record or the bit DSNBISCA of DSNBFLG1 for the DSNB record indicate the state. The data set is probably cataloged on more than one system. As soon as it is uncataloged on one of the MVS catalogs, the flag is turned off.

CLB— TMSSMF83 creates this record. A volume of this multivolume set, which has been cataloged, does not have the proper volume sequence number as indicated in the TMC. The volume was probably cataloged out of sequence or the volume sequence number in the TMC is incorrect. Verify the volume chaining of the volume set. Run TMSPTRS, if needed.

CLC— TMSSMF83 created this record. The previous volume of this multivolume data set that is cataloged was not chained to this volume in the TMC. The current volume is not the last of the volume set. Check the volume chaining of the volume set. Run TMSPTRS, if needed.

CLD— TMSSMF83 created this record. The current volume that is cataloged is not a part of the multivolume data set that is defined to the TMC. The volume is not in scratch status. If the catalog operation including this volume should add this volume to the volume chain, chain it manually.

CLE— TMSSMF83 created this record. A volume of this multivolume data set that is cataloged is not a part of this multivolume data set that is defined to the TMC. The volume is in scratch status but the TMERASE bit of TMFLAG3 in the TMC volume record is on. If the catalog operation including this volume should add this volume to the volume chain, erase the volume and chain the volume manually.

CLF— TMSSMF83 created this record. A volume of this multivolume data set that is cataloged is not a part of this multivolume data set that is defined to the TMC. The volume is in scratch status but it is also in delete status. If the catalog operation including this volume should add this volume to the volume chain, chain the volume manually.

CLG— TMSSMF83 created this record. A volume of this multivolume data set that is cataloged is not a part of this multivolume data set that is defined to the TMC. The volume is not being chained following the last volume of the volume set. If the catalog operation including this volume should add this volume to the volume chain, chain the volume manually.

CLH— TMSSMF83 created this record. A volume of this multivolume data set that is cataloged is not a part of this multivolume data set that is defined to the TMC. The volume is in scratch status but it is not in delete status. The volume should be in delete status because the previous volume in the volume chain has the TMACTVOLI flag of TMFLAG4 turned on. If the catalog operation including this volume should add this volume to the volume chain, chain the volume manually.

CLI— TMSSMF83 created this record. A catalog or recatalog request was being processed for the first file of a volume. The volume sequence number, on the TMC, was not equal to one. A recatalog request specifying multiple volumes for the first file in the wrong order causes this state.

CLJ— TMSSMF83 created this record. A catalog operation for a tape GDG caused more than 50 generations of the GDG data set to be uncataloged. CA 1 updates only the TMC records of the 50 oldest generations and turns off the OSC flag bits. The Audit record shows the last generation that was processed successfully. If any newer versions of the GDG data set that are no longer cataloged are required, update them manually.

EDA—This record was created when an External Data Manager called TMSTMSTV to release a tape volume. Either the volume is not flagged as externally managed, or it does not match any active EDM rule.

Verify the status of the volume and the EDM rules defined in PPOPTION member TMOEDMnn. If necessary, use TMSUPDTE to clear the EDM information and to expire the tape manually.

EDB—This record was created when an External Data Manager called TMSTMSTV to release a tape volume. The TMC volume was part of a multifile or multivolume chain, which is not valid for an EDM tape.

Verify the status of the volume and the EDM rules defined in PPOPTION member TMOEDMnn. If necessary, use TMSUPDTE to clear the EDM information and to expire the tape manually.

SCR— Invalid multivolume chain found.

BYP— A user exit bypassed or modified an update of a record. Original record with CA 1 modifications is written using BYP.

4

Tapes which have been rejected for output.

5

CA 1 internal processing records. Type 5 Audit records contain control statement or message input, the name of the function, and the CPU ID that wrote the record. The audit record contains the date and time when the processing record was written. The Type 5 Audit record codes are:

001— Extension (TMSFORMT) or purge (TMSREMOV) job reformatted the TMC.

002— CA 1 initialization control statement input or messages.

003— Detects whether CA 1 is at the proper level to support the TMSXTEND process.

009— The Blockset Audit feature uses this record to fill out an audit block.

010— The Blockset Audit feature uses this record to record that a dormant system has been detected and the block has been marked full.

011— Volume pool monitor record that updates volume status counters.

012— Volume pool monitor record that updates product variables that various CA 1 utilities created.

SORT Control Statement

The SORT control statement is optional and specifies the sort sequence for the Audit reports. The format for the SORT control statement is:

```
1    6

SORT=[value]
```

Control Statement Definition

SORT

Is the control statement identifier and must start in position 1.

value

Indicates the requested sort sequence. The valid sort values are:

VOLSER

Volume serial number. Audit record types 0, 1, 2, 3 and 4 are reported. Records are sorted by VOLSER, then by date/time.

DSN

Data set name. Only Audit record types 0, 1, 2, and 3 are reported. Records are sorted first by data set name, then by date/time.

TIME

Date and time the Audit records were logged. All record types can be reported. Records are sorted only by date/time. This is the default sort sequence.

TYPE

Audit record type code. The six type codes were listed previously under TYPE Control Statement (see page 101). Records are sorted first by type code, then by date/time.

DATE Control Statement

The DATE control statement is optional and is used to specify a single date or a range of dates to be selected. The format for the DATE control statement is:

```
1    6

DATE=yyyy/ddd[-yyyy/ddd]
```

Control Statement Definition

DATE

Is the control statement identifier and must start in position 1.

'yyyy/ddd'

A single date or range of dates to be selected. Use a dash (-) to indicate a range of dates. The date format *must* be consistent with the pattern optionally specified by the DATEFMT parameter on the EXEC statement, or in Julian date format (yyyy/ddd). In addition, dates that contain commas, blanks, or hyphens *must* be specified within delimiters to allow the program to correctly interpret the syntax of the statement; otherwise, a U0006 or S0C4 abend may occur. For example, DATE='MMMDD YYYY'-'MMMDD YYYY' must be delimited because the dates contain blanks.

TIME Control Statement

The TIME control statement is optionally used to specify a particular time or range of time to be selected. The format for the TIME control statement is:

```
1    6

TIME=hhmm[-hhmm][,GMT]
```

Control Statement Definition

TIME

Is the control statement identifier and must start in position 1.

hhmm

Indicates the time or range of time to be selected. Use a dash (-) to indicate a range of time. The value of hh (hours) can be 00 through 24; the value of mm (minutes) can be 00 through 59.

GMT

Indicates that the time or range of time to be selected is specified in GMT. For BLOCKSET audit files, the time stamps are in GMT/UTC. GMT audit files allow the audit to be shared across time zones. They also record HHMMSSth to provide the extra granularity that is needed for active audits.

DSN Control Statement

The DSN control statement is optional and is used to specify the data set name(s) to be selected. The format for the DSN control statement is:

```
1   5

DSN=dsn[-]
```

Control Statement Definition

DSN

Is the control statement identifier and must start in position 1.

dsn

Indicates the data set name to be selected. Use a dash (-) to select all data sets that begin with the specified characters.

FORMAT Control Statement

The FORMAT control statement is optional and is used to specify the print format for selected records. The format for the FORMAT control statement is:

```
1      8

FORMAT=[format]
```

Control Statement Definition

FORMAT

Is the control statement identifier and must start in position 1.

format

Specifies the print format. The values for format are:

SHORT

Only the most significant fields are printed. One line of data per Audit record appears on the report. This is the default.

LONG

All fields are printed, and nine lines of data per Audit record appears on the report.

DUMP

The record is printed in character and hexadecimal format.

Control Statement Examples

The following example shows the TMSAUDIT control statements necessary to generate a report on Type 0, 1, 2, 3 and 4 Audit records for volume serial numbers in the range 353527-353530. The records are sorted by VOLSER.

```
VOLSER=353527-353530
TYPE=0,1,2,3,4
SORT=VOLSER
```

The next example shows the TMSAUDIT control statements necessary to generate a report on all Type 0 Audit records for VOLSERs 013990 and 016512. The records are sorted by TIME.

```
VOLSER=013990,016512
TYPE=0
SORT=TIME
```

This example shows the TMSAUDIT control statements necessary to generate a report in LONG format for all Type 0, 1 and 2 Audit records associated with VOLSER 173543. The records are sorted by DSN.

```
VOLSER=173543
FORMAT=LONG
TYPE=0,1,2
SORT=DSN
```

This example shows the control statements necessary to generate a report of all Type 4 Audit records sorted by VOLSER.

```
TYPE=4
SORT=VOLSER
```

The final example shows the control statements necessary to generate a report in LONG format for Type 0 ,1, 2, and 3 Audit records between the Julian dates 2009/012 and 2009/224 associated with VOLSER 173567.

```
VOLSER=173567
FORMAT=LONG
TYPE=0,1,2,3
DATE=2009/012-2009/224
```

TMS Report 32 - Audit Data Set Activity

```
THURSDAY, MARCH 25,2005.207            Y O U R  C O M P A N Y  N A M E

TMS REPORT-32   REPORT TITLE

                             DEFINITION OF TMC FLAG-1 BITS
                                80 = INTERNAL FIELD CHANGED BY USER
                                40 = VOLUME CLOSED NORMALLY
                                20 = TMC RECORD UPDATED BY USER
                                10 = VOLUME CLOSED BY ABEND
                                08 = VOLUME LISTED TO BE CLEANED
                                04 = VOLUME LISTED IN SCRATCH STATUS
                                02 = VOLUME IN DELETE (INACTIVE) STATUS
                                01 = VOLUME ELIGIBLE FOR RDS OVERRIDE

                             DEFINITION OF TMC FLAG-2 BITS
                                80 = DATA SET WAS ON MVS CATALOG
                                40 = VOLUME OPENED FOR OUTPUT
                                20 = DATA SET RECREATED
                                10 = TEMPORARY DATA SET
                                08 = EXPIRED FROM CATALOG CONTROL
                                (TMSCTLG)
                                04 = EXPIRED FROM CYCLE CONTROL
                                (TMSCYCLE)
                                02 = EXPIRED FROM LDATE CONTROL
                                (TMSCLEAN)
                                01 = EXPIRED BY TMS, CA 11 OR EDM

                             DEFINITION OF TMC FLAG-3 BITS
                                80 = CA-9/R+ INDICATED BAD TAPE
                                40 = TAPE RELEASED BY EXT VAULT MANAGER
                                20 = CONTROLLED BY EXTERNAL DATA MANAGER
                                10 = CONTROLLED BY CA DYNAM/T
                                08 = DATA SET ERASE REQUIRED
                                04 = DEFAULT EXPDT USED AT OPEN
                                02 = MULTIPLE FILES EXIST
                                01 = CREATED BY CA 1/COPYCAT

                             DEFINITION OF TMC FLAG-4 BITS

                                80 = TAPE HAS BEEN EXPIRED BY SMS
                                40 = ACTUAL VOLSER IN USE
                                20 = VAULT SPECIFIC REQUEST
                                10 = TAPE HAS BEEN DEGAUSSED
                                08 = FILE ON OS CATALOG
                                04 = NON-RESIDENT TAPE
                                02 = TAPE IS IN USE FOR REALTIME STACKING
                                01 = NO FURTHER STACKING IS ALLOWED ON
                                TAPE

                              DEFINITION OF TMC FLAG-5 BITS

                                80 = TAPE IS USED FOR REALTIME STACKING
                                40 = VOLUME OWNED BY AN AGENT
                                20 = VIRTUAL VOLUME EXPIRED
                                10 = TAPE IS 3592 WORM CARTRIDGE
                                08 = WORM WWID AND WMC ARE PRESENT
```

TMS Report 32 - Batch Update Audit Records (Type 0)

```
THURSDAY, JANUARY 29, 2005.242             Y O U R   C O M P A N Y   N A M E                           ROxxxxx PAGE=00001

TMS REPORT-32   AUDIT RECORDS     TYPE=0                 FORMAT=SHORT
SORT=TIME



                                                                                 -------------- TIME STAMP AREA ----------------
VOLUME  VOL  FILE     DATA SET NAME
EXPDT     F F F   PROGRAM/    DATE    TIME  USERID   CPU  TYPE/
        SEQ  SEQ                                                       1 2 3     HOOK                                      CODE

980009  001       CAI.TEST.ACF2.CREATE                       YYYY/365    E14000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980009       002  TMS.TESTAGGR.FILE2                         YYYY/365    0100    TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980009  001       CAI.TEST.ACF2.CREATE                       YYYY/365    E14000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980009       003  TMS.TESTAGGR.FILE3                         YYYY/250    0100    TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980009  001       CAI.TEST.ACF2.CREATE                       YYYY/365    E14000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980009       004  TMS.TESTAGGR.FILE4                         YYYY/180    0100    TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980009  001       CAI.TEST.ACF2.CREATE                       YYYY/365    E14000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980010  001       CA1.TEST.FILE1                             CYCLE/015   414000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980011  002       CA1.TEST.FILE1                             CYCLE/015   414000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980013  003       CA1.TEST.FILE1                             CYCLE/015   414000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980013       002  CA1.TEST.FILE2                             2018/012    0100    TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980013  003       CA1.TEST.FILE1                             CYCLE/015   414000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980013       003  CA1.TEST.FILE3                             CATALOG     0100    TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980013  003       CA1.TEST.FILE1                             CYCLE/015   414000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980013       004  CA1.TEST.FILE4                             CYCLE/015   0100    TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980013  003       CA1.TEST.FILE1                             CYCLE/015   414000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980012       005  CA1.TEST.FILE5                             CATALOG     0100    TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980012  001                                                  YYYY/001    440000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980012  004       CA1.TEST.FILE1                             CYCLE/015   404000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980015  001       CA1.NLTEST.FILE1                           CYCLE/015   414000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980015       002  CA1.NLTEST.FILE2                           2018/012    0100    TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980015  001       CA1.NLTEST.FILE1                           CYCLE/015   414000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980015       003  CA1.NLTEST.FILE3                           CATALOG     0100    TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980015  001       CA1.NLTEST.FILE1                           CYCLE/015   414000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980014       004  CA1.NLTEST.FILE4                           CYCLE/015   0100    TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980014  001                                                  YYYY/001    440000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980014  002       CA1.NLTEST.FILE1                           CYCLE/015   404000  TMSMERGE YYYY/242   15.44 OPER001  XE90 BATCH
980020  001       THIS.IS.A.TEST                             YYYY/001    600000  TMSIONLN YYYY/243   09.55 USERID1  XE90 BATCH
980021  001       THIS.IS.ONLY.A.TEST                        YYYY/001    600000  TMSIONLN YYYY/243   09.55 USERID1  XE90 BATCH
980022  001       YOU.WOULD.KNOW.IF.THIS.WAS.REAL            YYYY/001    600000  TMSIONLN YYYY/243   09.56 USERID1  XE90 BATCH


              30 RECORDS
```

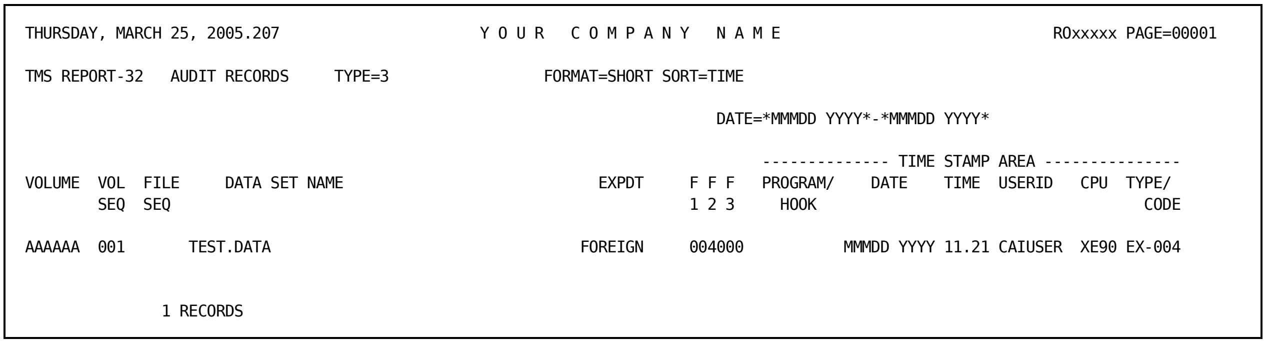
TMS Report 32 - Input Activity Audit Records (Type 1)

```
THURSDAY, MARCH 18, 2005.200                      Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-32   AUDIT RECORDS     TYPE=1                 FORMAT=SHORT SORT=VOLSER



                                                                                 -------------- TIME STAMP AREA ----------------
VOLUME  VOL  FILE     DATA SET NAME                            EXPDT     F F F   PROGRAM/    DATE    TIME  USERID   CPU  TYPE/
        SEQ  SEQ                                                         1 2 3     HOOK                                    CODE

980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  SMF 83   MMMDD YYYY 10.13 SYSPROG  XE90 INPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  SMF 83   MMMDD YYYY 10.36 SYSPROG  XE90 INPUT
980006  001       CAI.PROG.VERS.CREATE                       MMMDD YYYY  416000  SMF 83   MMMDD YYYY 15.46          XE90 INPUT
980007  001       CAI.PROG.VERS.CREATE                       MMMDD YYYY  416000  SMF 83   MMMDD YYYY 08.41          XE90 INPUT
980501  001       CAI.PROG.VERS.CREATE                       MMMDD YYYY  055000  SMF 83   MMMDD YYYY 14.34          XE90 INPUT


               5 RECORDS
```

TMS Report 32 - Output Activity Audit Records (Type 2)

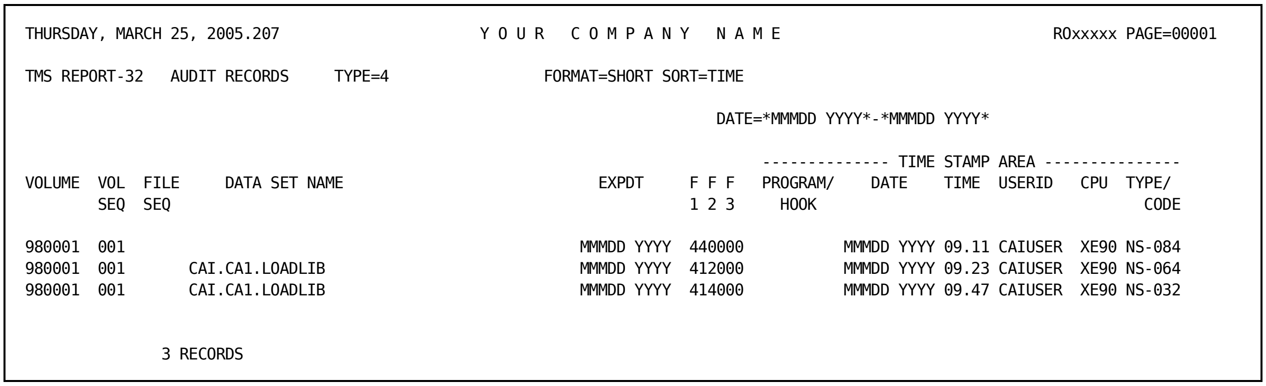
```
THURSDAY, MARCH 18, 2005.200                      Y O U R   C O M P A N Y   N A M E                   ROxxxxx PAGE=00001

TMS REPORT-32   AUDIT RECORDS     TYPE=2                 FORMAT=SHORT SORT=VOLSER



                                                                                 -------------- TIME STAMP AREA ----------------
VOLUME  VOL  FILE     DATA SET NAME                            EXPDT     F F F   PROGRAM/     DATE    TIME  USERID    CPU  TYPE/
        SEQ  SEQ                                                         1 2 3     HOOK                                      CODE

980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  014000  OP SL OT MMMDD YYYY 13.19 SYSPROG  XE90 OUTPUT
980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  414000  CLOSE IO MMMDD YYYY 13.19 SYSPROG  XE90 OUTPUT
980004  001       CAI.PROG.VERS.CREATE.TSS01                 MMMDD YYYY  014000  OP SL OT MMMDD YYYY 15.47 SYSPROG  XE90 OUTPUT
980004  001       CAI.PROG.VERS.CREATE.TSS01                 MMMDD YYYY  414000  CLOSE IO MMMDD YYYY 15.47 SYSPROG  XE90 OUTPUT
980004       002  CAI.PROG.VERS.TSS01                        MMMDD YYYY  0100    OP SL OT MMMDD YYYY 15.52 SYSPROG  XE90 OUTPUT
980004  001       CAI.PROG.VERS.CREATE.TSS01                 MMMDD YYYY  414000  OP SL OT MMMDD YYYY 15.52 SYSPROG  XE90 OUTPUT
980004       002  CAI.PROG.VERS.TSS01                        MMMDD YYYY  0100    CLOSE IO MMMDD YYYY 15.52 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  014000  OP SL OT MMMDD YYYY 09.22 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  CLOSE IO MMMDD YYYY 09.22 SYSPROG  XE90 OUTPUT
980005       002  CAI.PROG.VERS.TAPE02                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 09.24 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  OP SL OT MMMDD YYYY 09.24 SYSPROG  XE90 OUTPUT
980005       002  CAI.PROG.VERS.TAPE02                       MMMDD YYYY  0100    CLOSE IO MMMDD YYYY 09.24 SYSPROG  XE90 OUTPUT
980005       002  CAI.PROG.VERS.TAPE02                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 09.48 SYSPROG  XE90 OUTPUT
980005       003  CAI.PROG.VERS.TAPE03                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 09.48 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  OP SL OT MMMDD YYYY 09.48 SYSPROG  XE90 OUTPUT
980005       003  CAI.PROG.VERS.TAPE03                       MMMDD YYYY  0100    CLOSE IO MMMDD YYYY 09.48 SYSPROG  XE90 OUTPUT
980005       003  CAI.PROG.VERS.TAPE03                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 08.30 SYSPROG  XE90 OUTPUT
980005       004  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 08.30 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  OP SL OT MMMDD YYYY 08.30 SYSPROG  XE90 OUTPUT
980005       004  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    CLOSE IO MMMDD YYYY 08.30 SYSPROG  XE90 OUTPUT
980005       004  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 08.33 SYSPROG  XE90 OUTPUT
980005       005  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 08.33 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  OP SL OT MMMDD YYYY 08.33 SYSPROG  XE90 OUTPUT
980005       005  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    CLOSE IO MMMDD YYYY 08.33 SYSPROG  XE90 OUTPUT
980005       005  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 09.22 SYSPROG  XE90 OUTPUT
980005       006  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 09.22 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  OP SL OT MMMDD YYYY 09.22 SYSPROG  XE90 OUTPUT
980005       006  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    CLOSE IO MMMDD YYYY 09.22 SYSPROG  XE90 OUTPUT
980005       006  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 09.49 SYSPROG  XE90 OUTPUT
980005       007  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 09.49 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  OP SL OT MMMDD YYYY 09.49 SYSPROG  XE90 OUTPUT
980005       007  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    CLOSE IO MMMDD YYYY 09.49 SYSPROG  XE90 OUTPUT
980005       007  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 09.54 SYSPROG  XE90 OUTPUT
980005       008  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 09.54 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  OP SL OT MMMDD YYYY 09.54 SYSPROG  XE90 OUTPUT
980005       008  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    CLOSE IO MMMDD YYYY 09.54 SYSPROG  XE90 OUTPUT
980005       008  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 10.12 SYSPROG  XE90 OUTPUT
980005       009  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 10.12 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  OP SL OT MMMDD YYYY 10.12 SYSPROG  XE90 OUTPUT
980005       009  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    CLOSE IO MMMDD YYYY 10.12 SYSPROG  XE90 OUTPUT
980005       009  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 10.30 SYSPROG  XE90 OUTPUT
980005       010  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    OP SL OT MMMDD YYYY 10.31 SYSPROG  XE90 OUTPUT
980005  001       CAI.PROG.VERS.TAPE01                       MMMDD YYYY  414000  OP SL OT MMMDD YYYY 10.33 SYSPROG  XE90 OUTPUT
980005       010  CAI.PROG.VERS.CREATE                       MMMDD YYYY  0100    CLOSE IO MMMDD YYYY 10.36 SYSPROG  XE90 OUTPUT
```

TMS Report 32 - Audit Exception Records (Type 3)



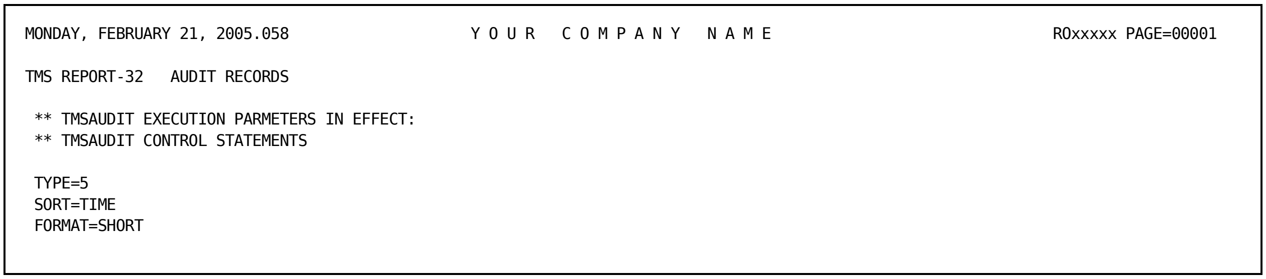
```
THURSDAY, MARCH 25, 2005.207                      Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-32   AUDIT RECORDS     TYPE=3                 FORMAT=SHORT SORT=TIME

                                                                            DATE=*MMMDD YYYY*-*MMMDD YYYY*

                                                                                 -------------- TIME STAMP AREA ---------------
VOLUME  VOL  FILE     DATA SET NAME                            EXPDT     F F F   PROGRAM/    DATE    TIME  USERID   CPU  TYPE/
        SEQ  SEQ                                                         1 2 3     HOOK                                    CODE

AAAAAA  001       TEST.DATA                                  FOREIGN     004000           MMMDD YYYY 11.21 CAIUSER  XE90 EX-004


               1 RECORDS
```

TMS Report 32 - Audit NOT SCRATCH Rejects (Type 4)



```
THURSDAY, MARCH 25, 2005.207                      Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-32   AUDIT RECORDS     TYPE=4                 FORMAT=SHORT SORT=TIME

                                                                            DATE=*MMMDD YYYY*-*MMMDD YYYY*

                                                                                 -------------- TIME STAMP AREA ---------------
VOLUME  VOL  FILE     DATA SET NAME                            EXPDT     F F F   PROGRAM/    DATE    TIME  USERID   CPU  TYPE/
        SEQ  SEQ                                                         1 2 3     HOOK                                    CODE

980001  001                                                  MMMDD YYYY  440000           MMMDD YYYY 09.11 CAIUSER  XE90 NS-084
980001  001       CAI.CA1.LOADLIB                            MMMDD YYYY  412000           MMMDD YYYY 09.23 CAIUSER  XE90 NS-064
980001  001       CAI.CA1.LOADLIB                            MMMDD YYYY  414000           MMMDD YYYY 09.47 CAIUSER  XE90 NS-032


               3 RECORDS
```

TMS Report 32 - Audit of Functional Modifications (Type 5)



```
MONDAY, FEBRUARY 21, 2005.058                    Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT-32   AUDIT RECORDS

 ** TMSAUDIT EXECUTION PARMETERS IN EFFECT:
 ** TMSAUDIT CONTROL STATEMENTS

 TYPE=5
 SORT=TIME
 FORMAT=SHORT
```

```
MONDAY, FEBRUARY 24, 2005.058                   Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-32   AUDIT RECORDS     TYPE=5                FORMAT=SHORT SORT=TIME


                                                                                  -------------- TIME STAMP AREA ----------------
CONTROL STATEMENT IMAGE/MESSAGE                                                   PROGRAM/    DATE    TIME  USERID   CPU  TYPE/
                                                                                    HOOK                                   CODE
UCB=-,                                                                00040000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
POOL=OTHER-TAPES                                                      00050000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
*                                                                                 TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
DSN=&.&TEMP.,                                                           00100000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
JOB=-,                                                                00110000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
UCB=E81,                                                              00120000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
POOL=TEMP-TAPEPOOL                                                    00130000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
*                                                                                 TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
DSN=-,                                                                00140000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
JOB=-,                                                                00150000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
UCB=E82,                                                              00160000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
POOL=PATTPOOL                                                         00170000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
*                                                                                 TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
DSN=ABCDE-,                                                           00180000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
JOB=ABCDE-,                                                           00190000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
UCB=A-,                                                               00200000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
POOL=TEST-TAPEPOOL                                                    00210000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
*                                                                                 TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
MGMTCLAS=INTERIM,                                                     00220000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
POOL=SMSPOOL                                                          00230000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
*                                                                                 TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
MGMTCLAS=SMSMGMT1,                                                    00240000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
POOL=TEST-TAPEPOOL                                                    00250000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
*                                                                                 TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
MGMTCLAS=SMSSMS,                                                      00260000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
POOL=TEST-TAPEPOOL                                                    00270000    TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
          TMSINITD PROCESSING FOR CTAPOPTN MEMBER TMOSCR00 COMPLETE               TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
          TMSINITD PROCESSING FOR CTAPOPTN MEMBER TMONSM00 COMPLETE               TMSINIT  YYYY/001   20.53 CAIUSER  XE90 05-002
```

TMS Report 32 - Selecting Record Types 0, 1, 2, 3 and 4

The following report reflects TMSAUDIT selecting Audit record Types 0, 1, 2, 3 and 4 and sorting by volume serial number, for volumes in the range 980000-980017. The output is in SHORT format.

```
THURSDAY, MARCH 10, 2005.200                      Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-32   AUDIT RECORDS     TYPE=0,1,2,3,4         FORMAT=SHORT SORT=VOLSER

 VOLSER=980000-980017

                                                                                 -------------- TIME STAMP AREA ---------------
VOLUME  VOL  FILE    DATA SET NAME                            EXPDT     F F F   PROGRAM/    DATE    TIME  USERID   CPU  TYPE/
        SEQ  SEQ                                                        1 2 3     HOOK                                    CODE

980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  014000  OP SL OT MMMDD YYYY 13.19 MILLS04  XE90 OUTPUT
980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  414000  CLOSE IO MMMDD YYYY 13.19 MILLS04  XE90 OUTPUT
980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  614000  TMSIONLN MMMDD YYYY 07.34 MILLS04  XE90 BATCH
980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  614000  TMSIONLN MMMDD YYYY 07.39 MILLS04  XE90 BATCH
980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  E14000  TMSIONLN MMMDD YYYY 13.12 MILLS04  XE90 BATCH
980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  E14000  TMSIONLN MMMDD YYYY 13.13 MILLS04  XE90 BATCH
980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  E54000  TMSIONLN MMMDD YYYY 13.13 MILLS04  XE90 BATCH
980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  E14000  TMSIONLN MMMDD YYYY 10.15 SMITH03  XE90 BATCH
980001  001       CAI.PROG.VERS.CREATEZ                      MMMDD YYYY  E14000  TMSIONLN MMMDD YYYY 11.11 SMITH03  XE90 BATCH
980001  001       DSNA.SETA                                  MMMDD YYYY  E14000  TMSIONLN MMMDD YYYY 15.01 ADMIN01  XE90 BATCH
980001  001       DSNA.SETA                                  MMMDD YYYY  E14000  TMSIONLN MMMDD YYYY 15.02 ADMIN01  XE90 BATCH
980001       002  SYSPROG.DSA02                              MMMDD YYYY  0000    TMSIONLN MMMDD YYYY 14.29 JONES01  XE90 BATCH
980001       003  SYSPROG.DSC.DS03                           CATALOG     0000    TMSIONLN MMMDD YYYY 14.29 JONES01  XE90 BATCH
980001       004  SYSPROG.DSC03                              CATALOG     0000    TMSIONLN MMMDD YYYY 14.29 JONES01  XE90 BATCH
980001       005  SYSPROG.DSA04                              CATALOG     0000    TMSIONLN MMMDD YYYY 14.29 JONES01  XE90 BATCH
980001  001       ADMIN01.DS01.MULTITMP                      CATALOG     600000  TMSIONLN MMMDD YYYY 14.29 JONES01  XE90 BATCH
980001       002  SYSPROG.DSA02                              CATALOG     0000    TMSIONLN MMMDD YYYY 09.12 JONES01  XE90 BATCH
980001       003  SYSPROG.DSC.DS03                           CATALOG     0000    TMSIONLN MMMDD YYYY 09.12 JONES01  XE90 BATCH
980001       004  SYSPROG.DSC03                              CATALOG     0000    TMSIONLN MMMDD YYYY 09.12 JONES01  XE90 BATCH
980001       005  SYSPROG.DSA04                              CATALOG     0000    TMSIONLN MMMDD YYYY 09.12 JONES01  XE90 BATCH
980001  001       ADMIN01.DS01.MULTITMP                      CATALOG     600000  TMSIONLN MMMDD YYYY 09.12 JONES01  XE90 BATCH
980001       002  SYSPROG.DSA02                              CATALOG     0000    TMSIONLN MMMDD YYYY 09.30 JONES01  XE90 BATCH
980001       003  SYSPROG.DSC.DS03                           CATALOG     0000    TMSIONLN MMMDD YYYY 09.30 JONES01  XE90 BATCH
980001       004  SYSPROG.DSC03                              CATALOG     0000    TMSIONLN MMMDD YYYY 09.30 JONES01  XE90 BATCH
980001       005  SYSPROG.DSA04                              CATALOG     0000    TMSIONLN MMMDD YYYY 09.30 JONES01  XE90 BATCH
980001  001       ADMIN01.DS01.MULTITMP                      CATALOG     600000  TMSIONLN MMMDD YYYY 09.30 JONES01  XE90 BATCH
980001       002  SYSPROG.DSA02                              CATALOG     0000    TMSIONLN MMMDD YYYY 09.40 JONES01  XE90 BATCH
980001       003  SYSPROG.DSC.DS03                           CATALOG     0000    TMSIONLN MMMDD YYYY 09.40 JONES01  XE90 BATCH
980001       004  SYSPROG.DSC03                              CATALOG     0000    TMSIONLN MMMDD YYYY 09.40 JONES01  XE90 BATCH
980001       005  SYSPROG.DSA04                              CATALOG     0000    TMSIONLN MMMDD YYYY 09.40 JONES01  XE90 BATCH
980001  001       SYSPROG.DSB01                              CATALOG     600000  TMSIONLN MMMDD YYYY 09.40 JONES01  XE90 BATCH
980001       002  SYSPROG.DSA02                              CATALOG     0000    TMSIONLN MMMDD YYYY 14.00 JONES01  XE90 BATCH
980001       003  SYSPROG.DSC.DS03                           CATALOG     0000    TMSIONLN MMMDD YYYY 14.00 JONES01  XE90 BATCH
980001       004  SYSPROG.DSC03                              CATALOG     0000    TMSIONLN MMMDD YYYY 14.00 JONES01  XE90 BATCH
980001       005  SYSPROG.DSA04                              CATALOG     0000    TMSIONLN MMMDD YYYY 14.00 JONES01  XE90 BATCH
980001  001       SYSPROG.DSB01                              CATALOG     600000  TMSIONLN MMMDD YYYY 14.00 JONES01  XE90 BATCH
980001  001       SYSPROG.DSB01                              CATALOG     600000  TMSIONLN MMMDD YYYY 07.48 SYSPROG  XE90 BATCH
980002  002       SYSPROG.DSB01                              CATALOG     600000  TMSIONLN MMMDD YYYY 09.40 JONES01  XE90 BATCH
980002  002       SYSPROG.DSB01                              CATALOG     600000  TMSIONLN MMMDD YYYY 14.00 JONES01  XE90 BATCH
980003  003       SYSPROG.DSB01                              CATALOG     600000  TMSIONLN MMMDD YYYY 09.40 JONES01  XE90 BATCH
980003  003       SYSPROG.DSB01                              CATALOG     600000  TMSIONLN MMMDD YYYY 14.00 JONES01  XE90 BATCH
980004  001       CAI.PROG.VERS.CREATE.TSS01                 MMMDD YYYY  014000  OP SL OT MMMDD YYYY 15.47 SYSPROG  XE90 OUTPUT
980004  001       CAI.PROG.VERS.CREATE.TSS01                 MMMDD YYYY  414000  CLOSE IO MMMDD YYYY 15.47 SYSPROG  XE90 OUTPUT
980004       002  CAI.PROG.VERS.TSS01                        MMMDD YYYY  0100    OP SL OT MMMDD YYYY 15.52 SYSPROG  XE90 OUTPUT
```

TMS Report 32 - Sample Dump

The following report reflects TMSAUDIT selecting Audit record Types 0, 1, 2, 3 and 4 and sorting by volume serial number, for volumes in the range 980000-980017. The output is in DUMP format.

```
THURSDAY, MARCH 10, 2005.200                      Y O U R   C O M P A N Y   N A M E                         ROxxxxx PAGE=00001

TMS REPORT-32   AUDIT RECORDS     TYPE=0,1,2,3,4         FORMAT=DUMP  SORT=VOLSER

 VOLSER=980000-980017

VOLSER=980001      980001CAI.PROG.VERS.CREATEZ                                                              T     &   &
                   FFFFFFCCC4ECEE4DCCC4CDCCECE44444444444444444444444092000000000000000000000000000000000400CE0900050005
                   980001319B3523B9136B395135900000000000000000000000006C0100000000000000000000000000001000032000000000
                  01...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                      <    SYSPROGBSTEP1   SYSUT2       <    SYSPROGB                 <
                   000000000000000009140039ECEDEFFCEECDF444EEEEEF440809140039ECEDEFFC0800000000000009140044444444444400
                   0000000C00000000006C011C693940122357100028243200E0006C011C69394012E000000000000C006C0100000000000000
                 101...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                 PROG.VERS.CREATEZ
                   00000000000000ECEE4DCCC4CDCCECE000000000000000000000000000000000000000000000000000000000000000000000
                   000000000000003523B9136B3951359000000000000000000000000000000000000000000000000000000000000000000000
                 201...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                             SYSPROG XE90            SYSPROG XE90
                   00000000000000000009140039ECEDEFF0ECFF000009140039ECEDEFF0ECFF00000000
                   000000000000000004006F011F6939401075902004006F011F69394010759020003000
                 301...5...10...15...20...25...30...35...40...45...50...55...60...65...70

VOLSER=980001      980001CAI.PROG.VERS.CREATEZ                                                              T     &   &
                   FFFFFFCCC4ECEE4DCCC4CDCCECE4444444444444444444444409200000000000000000000000000000000004400CE0900050005
                   980001319B3523B9136B395135900000000000000000000000006C0100000000000000000000000000001000032000000000
                  01...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                      <    SYSPROGBSTEP1   SYSUT2       <    SYSPROGB                 <
                   000000000000000009140039ECEDEFFCEECDF444EEEEEF440809140039ECEDEFFC0800000000000009140044444444444400
                   0001000C00000000006C011C693940122357100028243200E0006C011C69394012E000000000000C006C0100000000000000
                 101...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                 PROG.VERS.CREATEZ
                   00000000000000ECEE4DCCC4CDCCECE000000000000000000000000000000000000000000000000000000000000000000000
                   000000000000003523B9136B3951359000000000000000000000000000000000000000000000000000000000000000000000
                 201...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                             SYSPROG XE90            SYSPROG XE90
                   00000000000000000409140039ECEDEFF0ECFF000409140039ECEDEFF0ECFF00000000
                   000000000000000004006F011F6939401075902004006F011F69394010759020003000
                 301...5...10...15...20...25...30...35...40...45...50...55...60...65...70

VOLSER=980001      980001CAI.PROG.VERS.CREATEZ                                                         /    T     &   &
                   FFFFFFCCC4ECEE4DCCC4CDCCECE4444444444444444444444409200000000000000000000000000000000006400CE0900050005
                   980001319B3523B9136B395135900000000000000000000000006C0100000000000000000000000000001000032000000000
                  01...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....
```

TMSAUDIT Report Field Definitions

TMSAUDIT EXECUTION PARAMETERS IN EFFECT:

Parameters passed to program

TMSAUDIT CONTROL STATEMENTS

Input control statements

Control statement image

Input control statement

TYPE=x,x,x,x,x,x

Type or types of Audit records reported on

FORMAT=xxxx

Format of Audit records reported on

SORT=xxxxxx

Sort sequence of Audit records reported on

DATE=xxxxxxxxxx-xxxxxxxxxx

Date or date range reported on

TIME=xxxx-xxxx

Time or time range reported on

VOLSER=xxxxxx-xxxxxx

Volume serial or volume serial range reported on

DSN=dsname

Data set name or data set name prefix reported on

VOLUME

Volume serial number of tape volume or file

VOL SEQ

Volume sequence number of tape volume

FILE SEQ

File sequence number of file

DATA SET NAME

Data set name of tape volume or file

EXPDT

Expiration date of tape volume or file

F1

FLAG1 value of tape volume or file

F2

FLAG2 value of tape volume or file

F3

FLAG3 value of tape volume written

F4

FLAG4 value of tape volume or file

TIME-STAMP-AREA

PROGRAM/HOOK

Program or operating intercept ID causing Audit record to be written

DATE

Date that Audit record was written

TIME

Time that Audit record was written

USERID

User ID of job causing Audit record to be written

CPU

CPU ID where Audit record was written

TYPE/CODE

Type of record or code associated with Audit record written

RECORDS

Number of Audit records processed that are reported for this type

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

Use the TMSBINQ batch inquiry program to select and print volume, DSNB, and CA 1 control records.

CA 1 must be active or batch active.

Report Description

TMSBINQ produces TMS Report 12 - Batch Inquiry. The report contents are determined by the options selected: HEADERS, VOL, DSN or DSNB.

The HEADERS control statement generates header record output (Control Records from the TMC and Audit) that is formatted differently than output for nonheader records. Control Record reports are automatically generated in both character and hexadecimal format, as illustrated in the report samples on TMS Report 12 - Batch Inquiry (Headers) (see page 124).

Header records contain the following:

- the last CA 1 IPL date and time
- the number and values of (internal) volume serial number ranges
- the value of FLAG1 from Control Record #1
- the number of DSNBs allocated and used
- the DSNB base and the first available DSNB
- the number of allocated Audit records
- the first and next available Audit record
- the next label to be printed by TMSLBLPR
- the next CA Dynam/T pointer
- the date, time, volume, and data set name of the last TMC backup

The contents of Volume or DSNB records can be listed in three different formats: SHORT, LONG and DUMP. For greater clarity, the LONG format gives actual descriptions of the flag bytes and the fields that have special meanings, such as RECFM, TRTCH, DEN and BATCH ID.

Note: When the audit file is in the BLOCKSET mode, the audit dates and times are in GMT/UTC. All other dates and times are in local time.

Job Control Statements

```
//stepname EXEC PGM=TMSBINQ[,PARM='DATEFMT=(fmt)']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//SYSIN    DD  *
(control statements here)
/*
```

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 12. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

Completion Codes

Possible completion codes and their meanings are listed below.

0

Successful completion.

12

Error. See messages in TMSRPT DD in the Message Reference Guide.

Control Statement Specification

There are four control statements available for use with TMSBINQ:

- HEADERS is an optional control statement. It requests the TMC and Audit header records to be printed.
- VOL is an optional control statement. It requests a specific volume serial number or volume serial number range to be printed.
- DSN is an optional control statement. It requests a specific data set to be printed. The DSN must have an entry on the system catalog.
- DSNB is an optional control statement. It requests a specific DSNB or a DSNB range to be printed.

Multiple control statements can be entered for each program execution.

HEADERS Control Statement

The HEADERS control statement causes the three TMC header Control Records and the Audit header record to be printed in a hexadecimal and character format. For the TMC and Audit record definitions and keywords see the the *Programming Guide*. The format for the HEADERS control statement is:

```
1

HEADERS
```

Control Statement Definition

HEADERS

Is the control statement identifier and must start in position 1.

VOL Control Statement

The VOL control statement is used to inquire by volume serial number. If inquiry is made to a record containing DSNBs, the DSNBs associated with that volume are printed. If inquiry is made to a volume of a multivolume file, all subsequent volume serial numbers of the multivolume file are listed. However, if the VOL control statement requests a range of volumes, DSNBs and subsequent volumes are not printed. The format for the VOL control statement is:

```
1   5

VOL=vvvvvv[-vvvvvv][,format][,NEWPAGE]
```

Control Statement Definition

VOL

Is the control statement identifier and must begin in position 1.

vvvvvv

Is the volume serial number to be printed. If two volume serial numbers separated by a dash are specified, all TMC records included in the range are printed.

format

Specifies the print format. The values for format are:

SHORT

Only the most significant fields are printed.

LONG

All fields are printed. For each TMC record, nine lines of data will appear on the report. For each DSNB record, five lines of data will appear on the report. This is the default.

DUMP

The record is printed in character and hexadecimal format.

NEWPAGE

Forces a page break when multiple individual volumes or data set names are specified. If NEWPAGE is *not* specified, no page break occurs when the next control statement is processed. NEWPAGE is optional and, when used, must be the last option specified.

DSN Control Statement

The DSN control statement is used to inquire by data set name. The data set must have an entry on the system catalog. The format for the DSN control statement is:

```
1   5

DSN=dsname[,format][,NEWPAGE]
```

Control Statement Definition

DSN

Is the control statement identifier and must begin in position 1.

dsname

Is the complete data set name of the data set to be printed. GDGs can be designated by specifying the data set name and the relative generation number. The DSN must have an entry on the system catalog.

format

Specifies the print format. The values for format are:

SHORT

Only the most significant fields are printed.

LONG

All fields are printed. For each TMC record, nine lines of data will appear on the report. For each DSNB record, five lines of data will appear on the report. This is the default.

DUMP

The record is printed in character and hexadecimal format.

NEWPAGE

Forces a page break when multiple individual volumes or data set names are specified. If NEWPAGE is *not* specified, no page break occurs when the next control statement is processed. NEWPAGE is optional and, when used, must be the last option specified.

DSNB Control Statement

The DSNB control statement requests specific DSNBs to be printed. The format for the DSNB control statement is:

```
1    6

DSNB=n[-n][,format][,NEWPAGE]
```

Control Statement Definition

DSNB

Is the control statement identifier and must begin in position 1.

n

Is the one- to six-digit relative DSNB number to be printed. If multiple DSNB numbers separated by a dash (-) are specified, all DSNBs included in the range are printed.

format

Specifies the print format. The values for format are:

SHORT

Only the most significant fields are printed.

LONG

All fields are printed. For each DSNB record, five lines of data will appear on the report.This is the default.

DUMP

The record is printed in character and hexadecimal format.

NEWPAGE

Forces a page break when multiple individual volumes or data set names are specified. If NEWPAGE is *not* specified, no page break occurs when the next control statement is processed. NEWPAGE is optional and, when used, must be the last option specified.

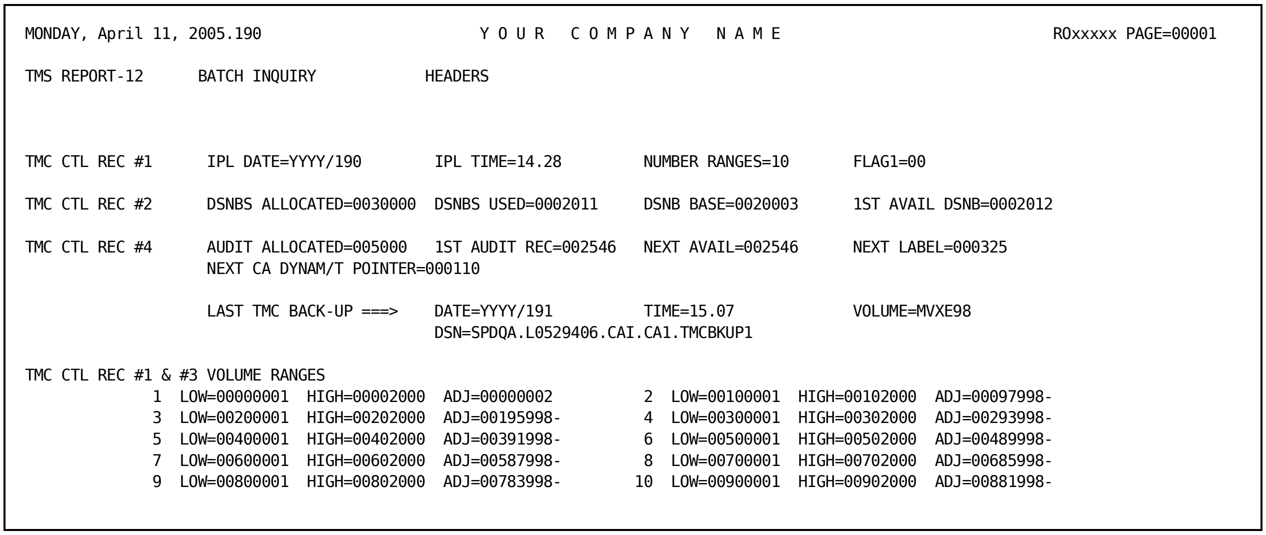
Control Statement Examples

The following example prints the SHORT format of the TMC volumes 000010, 000011 and 000012 and DSNB relative record 10. NEWPAGE will cause TMC volumes 000011 and 000012 to be printed on new pages. Volume 000101 are printed in the LONG format. Volume 000099 and the TMC volume containing the data set TMC.BACKUP are printed in hexadecimal, DUMP format. DSNB information for DSNB 000010 are printed in SHORT format.

```
VOL=000010-000012,SHORT,NEWPAGE
VOL=000101,LONG
VOL=000099,DUMP
DSNB=000010,SHORT
DSN=TMC.BACKUP,DUMP
```

TMS Report 12 - Batch Inquiry (Headers)

Header record output is generated in both character and hexadecimal format. The first page shows the character breakdown of the control Records. The second and third pages show the HEADERS output in hexadecimal format.



```
MONDAY, April 11, 2005.190                        Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-12      BATCH INQUIRY            HEADERS


TMC CTL REC #1      IPL DATE=YYYY/190        IPL TIME=14.28         NUMBER RANGES=10       FLAG1=00

TMC CTL REC #2      DSNBS ALLOCATED=0030000  DSNBS USED=0002011     DSNB BASE=0020003      1ST AVAIL DSNB=0002012

TMC CTL REC #4      AUDIT ALLOCATED=005000   1ST AUDIT REC=002546   NEXT AVAIL=002546      NEXT LABEL=000325
                    NEXT CA DYNAM/T POINTER=000110

                    LAST TMC BACK-UP ===>    DATE=YYYY/191          TIME=15.07             VOLUME=MVXE98
                                             DSN=SPDQA.L0529406.CAI.CA1.TMCBKUP1

TMC CTL REC #1 & #3 VOLUME RANGES
              1  LOW=00000001  HIGH=00002000  ADJ=00000002          2  LOW=00100001  HIGH=00102000  ADJ=00097998-
              3  LOW=00200001  HIGH=00202000  ADJ=00195998-         4  LOW=00300001  HIGH=00302000  ADJ=00293998-
              5  LOW=00400001  HIGH=00402000  ADJ=00391998-         6  LOW=00500001  HIGH=00502000  ADJ=00489998-
              7  LOW=00600001  HIGH=00602000  ADJ=00587998-         8  LOW=00700001  HIGH=00702000  ADJ=00685998-
              9  LOW=00800001  HIGH=00802000  ADJ=00783998-        10  LOW=00900001  HIGH=00902000  ADJ=00881998-
```

```
MONDAY, APRIL 11, 2005.190                        Y O U R   C O M P A N Y   N A M E                          PAGE=00003

TMS REPORT-12      BATCH INQUIRY            HEADERS


                   0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
                   0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
                 101...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....


                   0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
                   0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
                 201...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....


                   0000000000000000000000000000000000000000
                   0000000000000000000000000000000000000000
                 301...5...10...15...20...25...30...35...40

TMC CTL REC #4     TMSCTL#4          2          +               &"MVXE98SPDQA.L0529406.CAI.CA1.TMCBKUP1
                   EDECED7F00180000000F0073000D0042000D000009120057DEECFFEDCDC4DFFFFFFF4CCC4CCF4EDCCDEDF444444444444400
                   342333B400380000000920050007B00E3007C0000049C010F45759827481B30519406B319B311B34322471000000000000000
                  01...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                      2       >
                   000F000400060000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
                   00920015000E0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
                 101...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....


                   0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
                   0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
                 201...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....


                   0000000000000000000000000000000000000000
                   0000000000000000000000000000000000000000
                 301...5...10...15...20...25...30...35...40


       END OF REPORT 12
```

TMS Report 12 - Batch Inquiry (Long)

The following example illustrates that all files of a multifile data set are automatically generated.

```
THURSDAY, DECEMBER 12, 2013.346                 Y O U R   C O M P A N Y   N A M E
ROxxxxx PAGE=00001
TMS REPORT-12      BATCH INQUIRY            VOL=001508,LONG


001508  DSN=JESS.XPORT                                         ACCT=
        EXPDT=CATALOG      CDATE=2013/018    CTIME=13.33       CJOB=JESSSPLT
CSTEP=SPLTITUP   CDDNAME=EXPORT    CUNIT=078A
        FLAG1=E0=(INT,CLO,UPD)                                 FLAG2=60=(OUT,REC)
CPGM=TMSSPLIT
        FLAG3=06=(DEF,MTF)                                     FLAG4=00=
FLAG5=00=
        VOLSEQ=000001     1STVOL=         PREVVOL=      NEXTVOL=          ACTVOL=
DSN17=
        NUMDSNB=00001     1STDSNB=00000005  LSTDSNB=00000005                WWID=
WMC=
        LDATE=2013/018     LTIME=13.33       LJOB=JESSSPLT   LUNIT=078A
LPGM=TMSSPLIT                      BYTEPRC=188
        TRTCH=36TK       DEN=38KC       LABEL=SL      RECFM=FB        LRECL=000366
BLKSIZE=003660    BLKCNT=00000001
        OUTDATE=ZEROS      OUTCODE=          SLOT=0000000    BTHDATE=2013/018
COUNT=00003      VENDOR=            EDMID=
        TRERRI= 0000     TRERRC= 0000    TWERRI= 0000   TWERRC= 0000     VOLPERC=001
SMSMC=            ROBID=000
        PRERRI= 0000     PRERRC= 0000    PWERRI= 0000   PWERRC= 0000     FILPERC=001
COMPRES=000       ROBTY=
        DATECLN=ZEROS    USECLN=000000   CLNCNT=000    BATCHID=TMSUPDTE   HOOKID=
CTLGCNT=000       BESKEY=00000
        AUDATE=2013/346   AUTIME=16.03   USERID=CARPA20 CPUID=ZM1D      AUCODE=00
AUFLAG1=00
                                                 ***** DSNBS *****
001508  ACTIND=80       DSN=CARPA20.DSNBADD                             NEXTDSNB=00000000
PREVDSNB=00000000 CURDSNB=00000005
        EXPDT=CATALOG      VOLSER=001508    FILSEQ=00002     F1STVSN=001508
FLAG1=80=(UPD)                          FLAG2=00
        CDATE=2013/340     CTIME=15.17      CJOB=CARPABIN    CSTEP=TMSAGGR   CPGM=
FILPERC=000
        RECFM=          LRECL=00000     BLKSIZE=000000   BLKCNT=        BYTEPRC=070
BESKEY=00000     COMPRES=000
        BATCHID=TMSUDSNB   HOOKID=          BID=FF           SMSMC=
CTLGCNT=000
        AUDATE=2013/346   AUTIME=16.07   USERID=CARPA20  CPUID=ZM1D     AUCODE=00
AUFLAG1=00
       END OF REPORT 12
```

TMS Report 12 - Batch Inquiry (Short)

```
MONDAY, APRIL 11, 2005.190                        Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-12      BATCH INQUIRY            VOL=300009,SHORT


300009  DSN=CORP.TD.ARCHA                                    ACCT=
        EXPDT=YYYY/282      CDATE=YYYY/190    CTIME=14.32     CJOB=JOBARCHA       CSTEP=STEP1      CDDNAME=SEQOUT    CUNIT=0785

                                                 ***** DSNBS *****

300009  ACTIND=80          DSN=CORP.TD.ARCHB                                 NEXTDSNB=0000000 PREVDSNB=0000000 CURDSNB=0000001
        EXPDT=YYYY/282     VOLSER=300009    FILSEQ=002       F1STVSN=300009  FLAG1=01=(RDS)                            FLAG2=00


       END OF REPORT 12
```

TMS Report 12 - Batch Inquiry (Dump)

```
MONDAY, APRIL 11, 2005.190                        Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-12      BATCH INQUIRY            VOL=300009,DUMP


VOLSER=300009      300009CORP.TD.ARCHA                                                                      L
                   FFFFFFCDDD4EC4CDCCC4444444444444444444444444444444092200000000000000000000000000000044008D09000D0021
                   3000093697B34B193810000000000000000000000000000000048C01000000000000000000001000100011020032000700070
                  01...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                           JOBARCHASTEP1   SEQOUT            JOBARCHA
                   000400004444000009120042DDCCDCCCEECDF444ECDDEE440809120042DDCCDCCC080000000000000912004444444444444400
                   0005000C00000000049C013C16219381235710002586430075049C013C1621938175000000000000C049C020000000000000
                 101...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                 CORP.TD.ARCHA                     PGM111  PGM111
                   00000000000000CDDD4EC4CDCCC444400000000044444444DCDFFF44DCDFFF44000000000000000000000000000000000000
                   000000000000003697B34B19381000000000000000000000774111007741110000000000000000000000000000000000000
                 201...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                             CORPA01 XE98
                   00000000000000000109120042CDDDCFF0ECFF00
                   000000000000000004049F013F36971010759820
                 301...5...10...15...20...25...30...35...40

                                                 ***** DSNBS *****

  DSNB=000001                      300009300009  CORP.TD.ARCHB                                           JOBARCHASTEP2
                   F800000000000000FFFFFFFFFFFF00CDDD4EC4CDCCC4444444444444444444444444444444092209120042DDCCDCCCEECDF4
                   F0100001000000003000093000090236697B34B19382000000000000000000000000000000048C049C013C16219381235720
                  01...5...10...15...20...25...30...35...40...45...50...55...60...65...70...75...80...85...90...95.....

                                  3         PGM111                         CORPA01 XE98
                   4400000D0021000F944444444DCDFFF4400000000000000409120042CDDECFF0ECFF00
                   0000007000700003000000000774111000000000000000004049F013F36921010759820
                 101...5...10...15...20...25...30...35...40...45...50...55...60...65...70


       END OF REPORT 12
```

TMSBINQ Report Field Definitions

Control statement image

Input control statement

******* DSNBS *******

Related DSNB records display if first volume

TMC CTL REC #1

TMC Control Record #1 display

TMC CTL REC #2

TMC Control Record #2 display

TMC CTL REC #3

TMC Control Record #3 display

TMC CTL REC #4

Audit Control Record #4 display

TMC CTL REC #1 & #3 VOLUME RANGES

Volume ranges display from TMC Control Records #1 and #3

xxxxxx

Volume serial number SHORT or LONG display

VOLSER=xxxxxx

Volume serial number DUMP format display

DSNB=xxxxxx

DSNB number DUMP format display

TMSBLDUE Utility—Build TMSXITU and TMSXITE Definitions

Use the TMSBLDUE utility to assist in building TMMVOLDF statements for use with the CA 1 supplied exits TMSEXITU and TMSEXITE for an old format TMC. An old format TMC is one that is not converted by TMSXTEND. If you run TMSBLDUE against a new format TMC, one that is converted by TMSXTEND, the standard report is not produced. The utility ends with a good return code.

Important! With the introduction of the TMSXTEND utility in R11.0, users may no longer have to implement user exits TMSXITU and TMSXITE to provide support for alphanumeric volume serial numbers. If the TMSXTEND utility is run to create a new Extended Format TMC, volume definition records are added to the TMC and standard routines are employed to convert alphanumeric VOLSERs to the proper internal volume serial number.

TMSBLDUE evaluates the TMC, determines the type, and produces both a report describing how the TMC was formatted and what VOLSERs were found. The utility also generates TMMVOLDF statements to be assembled and link-edited to produce exits TMSXITU and TMSXITE. For both TMSBLDUE and TMMVOLDF to support a TMC, all VOLSERs must conform to certain rules:

1. Length of VOLSER must be four, five, or six characters.
2. 4- and 5-character VOLSERs can have blanks on left or right but not both.
3. Must be either all numeric or of the form AAANNN where AAA is a 1- to 5-character fixed alphanumeric prefix followed by a 1- to 5-character numeric portion. The alphanumeric portion must be made up from 0-9, A-Z or @$#.

TMSBLDUE will build a set of statements which reflect a definition of what is in the TMC. The statements may not be the only or the best definition or they may not correctly interpret unformatted areas that are reserved for future use. Review all output to ascertain the exit definitions and the requirements of your tape library.

The output from this utility (SYSPUNCH DD) should be used to update member TMSVOLDF of CAI.CTAPJCL and apply the TMSVLDF USERMOD. If USERMODs TMSXITE and TMSXITU (for user written exits) are already installed, they must be removed by doing an SMP RESTORE before installing TMSVOLDF.

Note: For information about the TMMVOLDF macro and the user exits, see the *Programming Guide*.

More information:

TMSXTEND Utility—Dynamic TMC Extend Process Control Program (see page 472)

Report Description

TMSBLDUE produces a report showing the TMMVOLDF statements generated and describes how the TMC was formatted.

Job Control Statements

```
//TMSBLDUE EXEC PGM=TMSBLDUE
//STEPLIB  DD DSN=CAI.CTAPLINK,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSPUNCH DD DSN=macros.fileout,DISP=(NEW,KEEP),    *Note
//            UNIT=SYSDA,
//            SPACE=(TRK,1),
//            DCB=(LRECL=80,BLKSIZE=3120,RECFM=FB)
//TMC      DD DSN=tmc.name,DISP=SHR
```

Note: The SYSPUNCH DD can be used to create an output file for use. The BLKSIZE can be any multiple of 80.

Control Statement Specifications

There are no control statements for TMSBLDUE.

Output Report

```
Processing input from 5.0/5.1 TMC
No parm specified, DEFAULT processing
Range 1 has internal vols 000000 to 003999
     for a total of 4000 Volumes
Total number of ranges: 1
         TMMVOLDF 000000,000999 Tmsformt range=0-999
         TMMVOLDF ABC000,ABC999 Tmsformt range=1000-1999,contiguous
         TMMVOLDF DEF000,DEF999 Tmsformt range=2000-2999,contiguous
         TMMVOLDF 003000,003999 Tmsformt range=3000-3999,contiguous
         END
TMC Records read: 4003
```

TMSBLDVR Utility—Build TMC Volume Ranges Table

The TMSBLDVR program is invoked by TMSXTEND to read the TMC and create a volume range table based on the ranges it finds, then adding or deleting any volume ranges or DSNBs as requested based on the control statements specified in the SYSIN DD. When executed in standalone mode, TMSBLDVR can be used to identify the volume ranges in an existing TMC, validate volume range or DSNB control statements, and determine the size requirement of the new TMC. The TMSBLDVR utility is a subprogram of TMSXTEND that can also be executed to create a new TMC, replacing functionality provided by the TMSFORMT utility.

TMSBLDVR can also be used to define a new, empty TMC. TMSBLDVR is superior to TMSFORMT for defining a new TMC because of its ability to format alphanumeric volume serial number ranges without the use of the TMSXITE and TMSXITU user exits. Also, when executed in standalone mode with the PARM=NEW parameter is the only time the TMCNEW DD statement is used.

Report Description

TMSBLDVR creates TMS REPORT-80 which displays the volume ranges found in the existing TMC. The report also lists the volume ranges that are built as a result of any add or remove range statements. Statistics are also produced for the number of total and free DSNBs. If you run TMSBLDVR in standalone mode without a PARM=NEW parameter, the report includes the size requirements for the new TMC based on a blocksize of either 340 or 8840. This feature helps ensure that the new TMC is correctly preallocated.

If the required size for the TMC exceeds 4,369 cylinders, allocate the TMC by adding DSNTYPE=LARGE to the DD statement in the allocation step. TMSPTRS does *not* support DSNTYPE=LARGE, use TMSAPEC instead of TMSPTRS if DSNTYPE=LARGE has been used.

The following messages are issued to the report 80 that comes out of TMSBLDVR:

UNIQUE TMC RNAME ACTIVATED

This message is displayed when UNIQUE_RNAME=YES is used to create a TMC and when extending an existing TMC. If you use the Exclude statement, you get an RC=08 error.

UNIQUE TMC RNAME DEACTIVATED

This message is displayed if UNIQUE_RNAME=NO is used when extending an existing TMC and unique RNAME was previously activated.

Job Control Statements

```
//STEP01     EXEC PGM=TMSBLDVR,REGION=0M [,PARM=NEW]
//TMSRPT80   DD SYSOUT=*
//TMCNEW     DD DSN=new.tmc.name,DISP=OLD,DCB=(BUFNO=60,BLKSIZE=n*340) * NOTE
//SYSUDUMP   DD SYSOUT=*
//*
//SYSIN      DD *
ADDVOL 500000-502999
ADDVOL VT0000-VT2999   ROBTY=88,VENDOR='CA VTAPE'
REMVOL TN0100-TN0599
/*
//
```

Note: This is an OPTIONAL DD statement, see PARM=NEW for details. The BLKSIZE may be equal to the LRECL or any multiple of the LRECL. However, when blocking the TMC the smallest valid block size is 1020. Also, when formatting a small blocked test TMC, ensure that it consists of at least 2 blocks.

Parameter Definitions

NEW

When this parameter is specified, a new TMC is formatted without copying volume ranges or DSNB records from an input TMC. The new TMC must be preallocated and cataloged. CA 1 does not have to be active or batch active to run with PARM=NEW.

JCL Considerations

SYSIN

This DD defines the control statement input data set. When run in standalone mode TMSBLDVR will validate that the function requested is proper for the requested range.

TMSRPT80

This is the report output DD for TMS REPORT-80. This DD statement is required.

TMCNEW

This DD statement is required if PARM=NEW is specified. A newly allocated, empty TMC should be specified.The TMC DCB attributes follow:

```
DCB=(RECFM=FB,LRECL=340,BLKSIZE=n*340,DSORG=PSU,BUFNO=60)
```

Completion Codes

The following completion codes are issued by TMSBLDVR:

0

Program completed successfully.

4

Program completed with minor errors. Review TMSRPT80 report for details on errors.

12

Program failed due to an invalid control statement.Review TMSRPT80 report for details on control statement errors.

Control Statement Specifications

ADDDSNB nnnnnnn

Requests that you add nnnnnnn free DSNB records to the new TMC.

ADDVOL

Lowvol-Highvol DEN=nn ROBTY=nn TRTCH=nn VENDOR=cccccccc DELETE specifies that the volume range lowvol-highvol are added to the new TMC. Optional parameters are provided to initialize the volume records with a density (DEN), robot type (ROBTY) recording technique (TRTCH), and vendor (VENDOR). For the DEN, ROBTY and TRTCH parameters, replace nn with hexadecimal values. For a list of these values, see TMC Volume Record Definition with Keywords in the *Programming Guide*. The DELETE parameter simply indicates that the volumes are added in DELETE status in the new TMC. VENDOR is a one-through-eight character string that is used to identify the media manufacturer. In addition to a range, you can specify a single VOLSER on the ADDVOL statement. If you do not use DELETE in the control statement, the added volumes are in SCRATCH status.

Note: If you use spaces in the VENDOR field, place single quotes around the value.

When you code an alphanumeric volser range, use the same alpha portion both in the beginning volser and in the ending volser.

Example: A00000-A99999 is valid but AA0000-AB9999 is invalid.

REMDSNB nnnnnnn

Requests that you remove nnnnnnn free DSNB records from the new TMC.

Note: REMDSNB is not valid with PARM=NEW and is mutually exclusive with the ADDDSNB control statement.

REMVOL lowvol-highvol

Specifies that all volumes in the range lowvol-highvol are removed from the TMC. All volumes in this range must be in DELETE status, otherwise the entire range is not removed.

Note: REMVOL is not valid with PARM=NEW.

When you code an alphanumeric volser range, use the same alpha portion both in the beginning volser and in the ending volser.

Example: A00000-A99999 is valid but AA0000-AB9999 is invalid.

UNIQUE_RNAME=[*YES*|*NO*]

Specifies if UNIQUE_RNAME is used with this TMC. If you code this control statement, code it as the first control statement in the SYSIN DD.

The following messages are issued to the report 80:

UNIQUE_TMC RNAME ACTIVATED

This message is displayed when UNIQUE_RNAME=YES is used to create a TMC and when you are extending an existing TMC.

UNIQUE_RNAME= MUST BE FIRST CONTROL STATEMENT

This message displays if UNIQUE_RNAME is not the first control statement coded. TMSBLDVR ends with the reason code 12.

TMS Report 80 - Report Field Definitions

```
TMS REPORT=80  JOB=MORTBLDV  STEP=TMSBLDVR  THURSDAY, APRIL 17,          2008.108 TIME=09:23:41
THURSDAY, APRIL 17, 2008.108   Y O U R   C O M P A N Y  N A M E                                                ROxxxxx PAGE=00001
TMSBLDVR      VOLUME DEFINITIONS

TMSBLDVR IS RUNNING STANDALONE MODE

TMC BEING READ: CAI.CA1.TMC

CURRENT TMC    VOLUME RANGES
EXTERNAL VSN   TMC REC NUMBER

120000-126999  0000004-0007003
D00001-D99999  0007004-0107002
HP0000-HP9999  0107003-0117002
H10000-H79999  0117003-0187002
KR0000-KR9999  0187003-0197002
KS0000-KS9999  0197003-0207002
K00001-K99999  0207003-0307001
O00001-O50000  0307002-0357001
P00001-P02500  0357002-0359501
TA0000-TA9999  0359502-0369501
TB0000-TB9999  0369502-0379501
TC0000-TC9999  0379502-0389501
```

External VOLSER Range

This field shows the low and high VOLSER numbers for a range found in the TMC.

TMC Rec Number

This field shows the low and high relative record number of the volume serial numbers for each range in the TMC.

In addition, TMS REPORT-80 also lists the TMC name being evaluated. The input control statements specified and the ranges that would be created for the new TMC. It also gives the number of DSNB's currently allocated, currently unused, and how many can be removed. Finally, it gives the space requirements of the TMC based on a unblocked TMC or a TMC allocated with a blocksize of 8840 (this is only reported if TMSBLDVR is run in standalone mode).

TMSCKDTM Utility—Verify Packed Fields in the TMC

Use the TMSCKDTM to verify that packed decimal fields in the TMC, such as date and time, are valid.

This utility should be run whenever an S0C7 occurs due to a bad date or time field, or an invalid date is displayed by other CA 1 utilities or online interfaces. An invalid date is displayed as ten asterisks (**********) on a batch report or by an online interface.

Once a TMC record with an invalid date or time is identified, it is important to determine how the TMC was updated with the bad value. TMSAUDIT can be run to list updates that occurred for the Volume or DSNB record with the bad date or time field (see TMSAUDIT in the this guide). Once the source of the bad update is found, the incorrect update by the interface should be corrected.

CA 1 can be active or batch active.

Report Description

TMSCKDTM produces TMS Report 15 - Invalid Date and Packed Fields. The volume serial or DSNB number is listed, the data set name, the field name containing the invalid value, the invalid value, and a message pertaining to the invalid value.

Job Control Statements

```
//stepname EXEC PGM=TMSCKDTM
//STEPLIB  DD DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD SYSOUT=A
```

Control Statement Specification

There are no control statements for TMSCKDTM.

TMS Report 15 - Invalid Date and Packed Fields

```
FRIDAY, FEBRUARY 11, 2005.054                 Y O U R   C O M P A N Y   N A M E                          ROxxxxx PAGE=00001

TMS REPORT-15 INVALID DATE AND PACKED FIELDS

VOLSER  REC TYPE DATA SET NAME                                FIELD    VALUE                MESSAGE

0000005 DSNB     CAI.CA1.DATA.FILE2                           CTIME   X'012040'             INVALID PACKED FIELD
0000007 DSNB     CAI.CA1.TEST.FILE3                           CTIME   X'016140'             INVALID PACKED FIELD
0000009 DSNB     CAI.CA1.MVMF.DATA.FILE3                      CTIME   X'014540'             INVALID PACKED FIELD
0000011 DSNB     CAI.CA1.TAPER.OU01SL02                       CTIME   X'023040'             INVALID PACKED FIELD

         END OF REPORT 15  -  TOTAL ITEMS LISTED=        4
```

TMSCKDTM Report Field Definitions

VOLSER

Volume serial number of tape volume or file

REC TYPE

Specifies whether the record is a volume or DSNB record

DATA SET NAME

Data set name of tape volume or file

FIELD

Field name of record which is in error

VALUE

Hexadecimal value of the invalid field

MESSAGE

Indicates the type of error

TMSCKLVL Utility—Print and Verify Maintenance Levels of Resident Tape Processing Modules

Use the TMSCKLVL utility to report CA 1 maintenance levels of key processing modules.

You may be asked to run this report to verify the levels of certain modules when contacting CA 1 support.

CA 1 can be active or batch active. TMSCKLVL can also be run while CA 1 is inactive. However, the TMS00SVC CSECTS will not be reported. A message is issued indicating that CA 1 is inactive.

Report Description

TMSCKLVL produces TMS Report 74 - Check Levels of CA 1 LPA and Link List modules. The current status of CA 1, the WTO subsystem, Realtime User Exits installed, anchored modules used in batch and realtime mode, and the CA 1 SVC is listed.

Job Control Statements

```
//stepname EXEC PGM=TMSCKLVL
//STEPLIB  DD DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD SYSOUT=A
```

Parameter Definitions

There are no parameters for TMSCKLVL.

Control Statement Specifications

There are no control statements for TMSCKLVL.

TMS Report 74 - Check Levels of CA 1 LPA and Link List Modules

```
TUESDAY, APRIL 30, 2013.120                      Y O U R   C O M P A N Y   N A M
E                                P02138  PAGE=00001
TMS REPORT-74                                 CHECK LEVELS OF CA 1 LPA AND LINKLIST
MODULES





CA 1 IS
ACTIVE

CA 1 OSI TMSOCE1A IS
INSTALLED

WTO SUBSYSTEM IS
ACTIVE

SSVT     ADDR =
00C3F230

CTSMSGVT ADDR = 9165D340 MAINT =
CBAFC60

DFARELS  ADDR =
03020100

SYSTEM   NAME =
XE90

RESIDENT
MODULE:

TMSTMVT  IS LOCATED AT 00BF5000 IN CSA - LEVEL = TVTC600 FLAG0=02 MAINT =
P02042
ADDRESSES STORED IN
TMVT:

TMSXITA  ADDR =   N/A    MAINT =         DATE =          TIME
=
TMSXITB  ADDR =   N/A    MAINT =         DATE =          TIME
=
TMSXITC  ADDR =   N/A    MAINT =         DATE =          TIME
=
TMSVOLU  ADDR = 80C0B350 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.27
TMSVOLE  ADDR = 80BF0850 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.27
CTSUXEDM ADDR =   N/A    MAINT =         DATE =          TIME
=
TMSXITS  ADDR =   N/A    MAINT =         DATE =          TIME
=
TMSXITF  ADDR =   N/A    MAINT =         DATE =          TIME
=
CTSXITG  ADDR =   N/A    MAINT =         DATE =          TIME
=
TMSXITJ  ADDR =   N/A    MAINT =         DATE =          TIME
=
TMSVOLU  ADDR = 80C0B350 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.27
TMSVOLE  ADDR = 80BF0850 MAINT = CAL0C60 DATE = 02/18/11 TIME
```

```
=  17.27
CTSRLS   ADDR = 80B96F10 MAINT = CBAFC60 DATE = 02/18/11 TIME
=  12.14
TMSRULS  ADDR = 80BAE280 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.25
TMSVOLDF ADDR = 00000000 MAINT =   N/A   DATE =   N/A    TIME
=   N/A
TMSVPMAD ADDR = 916C7070 MAINT = P02072  DATE = 02/27/13 TIME
=  14.42
CTSDATE  ADDR = 80BA9F28 MAINT =
CBAFC60

CTSSMS   ADDR = 902EDCD0 MAINT =
P00036

TMSLABEL ADDR = 80BA66B0 MAINT =
CAL0C60

TMSOCEPR ADDR = 902E8A20 MAINT =
P01995

TMSSECUR ADDR = 00BA2B88 MAINT =
P01961

TMSTVXCP ADDR = 9192E0C0 MAINT =
CAL0C60

TMSTVHSH ADDR = 00C34208 MAINT =
CAL0C60

TMSTVABD ADDR = 916FE5B0 MAINT =
P02025

TMSTVSNP ADDR = 9184C948 MAINT =
CAL0C60

TMSKEYXB ADDR = 902E1870 MAINT =
P02025

CTSCPOOL ADDR = 9184C0C0 MAINT =
CBAFC60

CTSEDMLK ADDR = 916FD790 MAINT =
CBAFC60
```

```
TUESDAY, APRIL 30, 2013.120                      Y O U R   C O M P A N Y   N A M
E                                         PAGE=00002
TMS REPORT-74                                  CHECK LEVELS OF CA 1 LPA AND LINKLIST
MODULES




CTSMSGPR ADDR = 901161E0 MAINT =
P00138

CTSMSGRT ADDR = 90114828 MAINT =
P00084

CTSMSGSP ADDR = 916C4570 MAINT =
CBAFC60

CTSWTX   ADDR = 917A2070 MAINT =
CBAFC60

CTSPARSE ADDR = 80B9A7A0 MAINT =
CBAFC60

CTSVTS00 ADDR = 00B98ED0 MAINT =
CBAFC60

CTSOCE00 ADDR = 00B9CFF8 MAINT =
CBAFC60

CTSPM    ADDR = 80B9F000 MAINT =
CBAFC60

TMSQSTS  ADDR = 00B9E080 MAINT =
P01904

TMSSB01  ADDR = 90112EF8 MAINT =
CAL0C60

TMS0MODV ADDR = 90110CC0 MAINT =
CAL0C60

MODULES FROM
LINKLIST:

TMSKEYAB ADDR = 00041000 MAINT = P02025  DATE = 04/16/13 TIME
=  14.09
TMSSVC   ADDR = 0000E0F0 MAINT = P02025  DATE = 11/04/11 TIME
=  17.17
TMSSRVCA ADDR = 0001B000 MAINT =   1104  DATE = 12/21/11 TIME
=  09.44
TMSSMF83 ADDR = 9010BC50 MAINT = P02110  DATE = 11/09/12 TIME
=  11.41
CBRUXEJC ADDR = 91F467D0 MAINT =   IBM   DATE = 11/15/12 TIME
=   N/A
CBRUXVNL ADDR = 9274F000 MAINT =   IBM   DATE = 11/15/12 TIME
=   N/A
CBRUXENT ADDR = 91F467D0 MAINT =   IBM   DATE = 11/15/12 TIME
=   N/A
TMS00SVC
CSECTS:

TMS0XEP  ADDR = 902F4088 MAINT = P02025  DATE = 11/04/11 TIME
```

```
=  17.17
TMS0XRT  ADDR = 102F5388 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.28
TMS0XEX  ADDR = 102F5B20 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.27
TMS0XF1  ADDR = 102F6180 MAINT = P02097  DATE = 11/14/12 TIME
=  11.09
TMS0XDT  ADDR = 102F79D0 MAINT = P02040  DATE = 11/08/11 TIME
=  16.26
TMS0XSC  ADDR = 102F8450 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.28
TMS0XEV  ADDR = 102F8870 MAINT = P02096  DATE = 08/15/12 TIME
=  14.05
TMS0XCU  ADDR = 102F9208 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.27
TMS0XSP  ADDR = 102F9998 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.28
TMS0XF2  ADDR = 102F9DA8 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.27
TMS0XAD  ADDR = 102FAFC8 MAINT = CAL0C60 DATE = 02/18/11 TIME
=  17.27
TMS0X98  ADDR = 102FB7C0 MAINT = P01491  DATE = 12/06/11 TIME
=  15.06
TMS0XCL  ADDR = 102FC6E8 MAINT = P02082  DATE = 04/27/12 TIME
=  08.39
TMS0YEP  ADDR = 102FDFC0 MAINT = P02072  DATE = 02/27/13 TIME
=  14.42
TMS0YIO  ADDR = 102FFAA8 MAINT = P02025  DATE = 11/14/11 TIME
=  09.26
TMS0YIN  ADDR = 10301678 MAINT = P02025  DATE = 11/04/11 TIME
=  17.17
TMS0YRT  ADDR = 103031A8 MAINT = P02040  DATE = 11/08/11 TIME
=  16.26
TMSESTAE ADDR = 103038B8 MAINT = P02025  DATE = 11/04/11 TIME
=  17.16
```

TMSCLEAN Utility—List Scratch, Clean and Scratch Inventory Volumes

Use the TMSCLEAN multipurpose program to perform scratch functions based on parameters specified in the JCL.

TMSCLEAN reads the complete TMC or specified ranges of volumes looking for eligible volumes (expired volumes) and passes the volsers to a subroutine called TMSSCR, which performs the scratch process. TMSSCR can be called from the online interface (for immediate scratch requests).

As the TMC is read sequentially, only volume records with a VOLSEQ=1 are examined. If the volumes are eligible for scratch processing, all secondary volumes and secondary files are processed. If chain errors are detected, a warning message is issued and the entire chain is bypassed, preventing invalid scratching of a tape with a chain error.

The TMSCLEAN utility enables you to select a subset of volumes for processing based on a volume range or a data set name pattern that is matched against file 1 on a volume. Volume range processing does not change how volumes are processed, once selected. All existing functions of TMSCLEAN can be used in conjunction with range processing.

Range control statements are specified in the SYSIN DD, which has standard SYSIN attributes (80 byte fixed length records, either blocked or unblocked). If the SYSIN DD is not found, TMSCLEAN processes the entire TMC. If the SYSIN DD is found with valid range control statements, only the volumes that fall in the specified ranges are processed. If a volume range is specified and a FILE1DSN is also specified, only data sets in the range that also match the FILE1DSN pattern are processed.

The function of TMSCTLG is moved to TMSCLEAN if the system option OCTLG is set to **N** (the default is **Y**). This option indicates if the old TMSCTLG program should be used or the new catalog processing in TMSCLEAN. The **old** TMSCTLG program should be used if the TMC is shared with a system not yet active on Version r5.2 or higher. Once all systems that share the TMC have been upgraded to r5.2 or higher and the TMSOSCAT utility has been run with the parameter of SYNC, the OCTLG parameter can be changed to **N**. After CA 1 has been reinitialized to make the change in options take effect, the next time TMSCLEAN runs it uses the new catalog process. At this time, the TMSCTLG utility no longer needs to be run.

Note: This option has no effect on the realtime interface to the OS catalog. It indicates if TMSCLEAN may trust the new catalog indicators maintained by the realtime interface. If all systems that share the TMC have not yet been upgraded to Version r5.2 or higher, and have not all had the TMSOSCAT utility executed with a PARM=SYNC; the catalog indicators should not be trusted and the old TMSCTLG utility should continue to be executed. Once the old version of TMSCTLG no longer needs to be executed, the OCTLG option can be changed to N.

Catalog processing examines the OS catalog indicator for every file under Catalog Control if the primary volume meets one of the following:

- The primary volume has an expiration date of CATALOG
- The primary volume has an expiration date of permanent and was created by CA 1 Copycat

If either condition is met, all secondary data sets are examined and processing does not stop until the end of the DSNB chain has been reached. Any file found under Catalog Control with the OS catalog indicator OFF has its expiration date changed to the current date plus the R9 value if one of the following conditions is met.

- The file has been previously cataloged
- The file is at least 24 hours old (CDATE is less than current date minus 1)

Note: There is still the potential for changing the expiration date of an active file. If your site has long-running jobs where a single-step runs more than 24 hours, and you wish to use Catalog Control to manage the retention of files created by this job, use the CATLG/ddd retention method where ddd is the longest time (plus 1) the job can run. For example, if a job runs for 5 days, CATLG/005 should be used as the retention (90006 in the JCL). This prevents the premature change in retention while the file is still being created but not yet cataloged at step-end.

The UNCATA system option controls the uncataloging of data sets. YES specifies to uncatalog all files when they are scratched. G specifies to uncatalog all non GDG files when they are scratched. NO specifies that no uncatalog operations take place.

The SCRCAT system option indicates if any file currently cataloged should be scratched (even if it is not Catalog Controlled). This option only applies to TMSCLEAN and not when the scratch program is executed using the online interface or the TMSARCTV program. If SCRCAT is set to Y (the default), the current catalog status is not used to determine if a volume can be scratched. If set to G (for GDG), GDG files are not scratched if they are still cataloged (regardless of whether the retention has been met). If set to N or NO, any file currently cataloged is not scratched. If this option is set to N or G, and any secondary file has been found to still be cataloged, the volume chain is not scratched. This option is ignored if the OCTLG option is set to Y.

Files controlled by LDATE/ddd or CATLG/ddd have their expiration dates changed (if they pass the ddd days test) even when they reside at an off-site location. However, they are not scratched until they have been returned from the off-site location. This allows for the correct rotation of tapes from off-site locations when the EXP keyword is used and the files are controlled with either LDATE/ddd or CATLG/ddd keyword values.

Control Parameters

The following JCL parameters control the various functions of the TMSCLEAN program.

- The *SCRATCHLIST* parameter marks expired tapes available for use and produces a flat file which is used to produce the Scratch Listing. Although a tape volume has expired, the volume cannot be used for output until it has appeared on the Scratch Listing and the X'04' bit is on in FLAG1. The following tapes are bypassed from SCRATCHLIST processing:

  - Tapes which are out-of-area
  - 3592 WORM tapes (excluding virtual WORM tapes)
  - Tapes under CA Dynam/T control
  - Tapes under control of an External Data Manager unless expired through the EDM interface.

  This flat file contains volume serial numbers of tapes that have come into scratch status by the execution of TMSCLEAN. Tapes can also be flagged for erasure. Multivolume chaining fields are cleared as volumes are set to scratch status. DSNB records will have the active indicator turned off and the DSNB is returned to the free queue. Some fields are not reset until the volume and DSNB are used again for output.

  A flat file, defined by the EARLOUT DD, is produced for this option and the other TMSCLEAN options. For SCRATCHLIST processing, this file can be used to uncatalog data sets on other systems. This is useful in an environment where system catalogs that contain tape data set entries are not shared or available to the system where TMSCLEAN is executed. TMSCLEAN may attempt to uncatalog data sets contained on volumes being scratched. When system catalogs are not available to the system on which TMSCLEAN is running, the TMC and OS catalogs will not be synchronized. To use this flat file, save the data set defined to the EARLOUT DD and use the utility TMSUNCAT to perform the uncatalogs on other systems. For more information on TMSUNCAT, see the TMSUNCAT utility.

  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 and updates the X'02' bit in FLAG2 to indicate the volume has been expired by Days Since Last Used Control.

  Data sets under Catalog Days Control are processed by the TMSCLEAN SCRATCHLIST option. Data sets with the Catalog Days Control (CATLG/ddd in control statements and online interfaces, or LABEL=EXPDT=90ddd in JCL) expiration date are held for ddd number of days, based on the current system date and the creation date of the data set. Once ddd days have been met, TMSCLEAN modifies the expiration date of the TMC Volume or DSNB Record to Catalog Control.

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. The flat file produced by TMSCLEAN SCRATCHLIST processing contains the appropriate scratch tape pool for the tapes being scratched. The scratch tape pool can be listed on a report produced by a report writer such as CA Earl.

Several CA 1 system options (defined in the TMOOPTxx member in *hlq*.CTAPOPTN) can invoke additional SCRATCHLIST processing. If the option DSEALL is set to YES all tape volumes being scratched are flagged as requiring data set erase (DSE) before they can be used in an output scratch request. The X'08' bit is turned on in FLAG3 of the TMC record. Any attempt to use the tape with this bit on is rejected with a NOT SCRATCH 22 message. The utility TMSTPPRO is used to perform the data erasure, and is the only method available to turn off the erasure required bit.

Another set of CA 1 system options can determine whether SMS Management Class maximum retention rules are to be applied to any nonscratch tape in the TMC. If the option SMS is set to YES (indicating that the CA 1 interface to SMS is active), and the SR option is set to YES (enforce SMS), each tape data set that contains a valid SMS Management Class will have its expiration date compared to its Management Class definition. When a tape volume exceeds the maximum retention defined by SMS, it is expired the following day, regardless of the value in the EXPDT field. A warning message is issued on the TMSCLEAN message report (TMS Report 85). Any tape expired in this manner will have the X'80' bit in FLAG4 turned on to indicate it has been expired by SMS Management Class maximum retention enforcement.

The CA 1 keyword date USER/uuu can also be processed by the SCRATCHLIST option for TMSCLEAN. User control (USER/uuu in control statements and online interfaces or LABEL=EXPDT=88uuu in JCL) are user-defined keyword dates, and are treated in the same manner as permanent hold. With the CA 1 option UXCLN2 set to the name of the exit itself, the user exit TMSXCLN2 is invoked for each TMC Volume record containing the USER/uuu expiration date.

Note: For more information on the TMSXCLN2 user exit see the *Programming Guide*.

- The *SCRATCHRELIST* parameter produces a flat file containing an inventory of all active tapes in scratch status, regardless of the length of time they have been in scratch status. The flat file can be used in a subsequent step with a report writer to generate the Scratch and Clean Relist. No additional scratching is performed with this parameter.
- The *EXTEND* parameter extends the expiration dates of tapes expiring on a specific date or within a range of days, such as company holidays or vacation shutdowns. The user exit TMSXCLN1 can optionally be invoked to modify or bypass TMC Volume records being extended.
- The *RESET* parameter scans the TMC and turns off the scratch bit (which had been turned on for records with a future expiration date) caused by a bad IPL date. The user exit TMSXCLN1 can optionally be invoked to modify or bypass TMC Volume records being reset.

- The *RESETCAT* parameter scans the TMC and resets the expiration date of Volume and DSNB records inadvertently expired by Catalog Control. The expiration date in the Volume and DSNB record is reset to the CA 1 keyword date CATALOG, and the expired Catalog Control indicator is turned off.

  If the old TMSCTLG program is no longer used (all systems sharing the TMC have been upgraded to r5.2 or higher, the OCTLG system parameter is set to N, and the TMSOSCAT utility has been run on all systems with the SYNC parameter), the TMSOSCAT utility should be rerun with the SYNC parameter to ensure the correct setting of the TMISCAT and DSNBISCA flags. These flags are not reset by executing TMSCLEAN with the RESETCAT option. Only the EXPDT and the TMECATLG FLAG2 x'08' bit are reset when RESETCAT is used. The user exit TMSXCLN1 can optionally be invoked to modify or bypass TMC volume records being reset. This method of recovery must be run before TMSCLEAN scratches the data sets.

- The *TEST* parameter may be used with any of the above options to prevent updating of the TMC. If present, TEST **must** be the first parameter specified. This option still produces the flat file which can be used to produce reports, but suppresses any updating of the TMC. If TEST is used, the Tapes scratched today count is off by the total number of secondary volumes under LDATE control or Catalog Control if R9 is set to zero and tapes will **not** be on the CA Earl reports.

  Note: The user exit TMSXCLN2 is only invoked if the UXCLN2 option is set to the name of the exit itself. The TMSXCLN1 exit is only invoked if the UXCLN1 option is set to the name of the exit itself.

General Information

Each TMSCLEAN option produces a flat file of records processed in the CA 1 Retention Record format. The flat file is defined by the EARLOUT DD, and can be used in a subsequent step with a report writer such as CA Earl to produce customized reports. Sample CA Earl report members are provided in the CA 1 Source library, and can be modified to produce listing in any sequence and listing any fields required.

The CA 1 Retention Record format is described in the *Programming Guide*, and is a TMC Volume or DSNB record followed by an extension that contains fields such as original expiration date, tape volume cleaning information, tape scratch pool names, uncatalog information, and security return codes.

Tapes owned by CA Dynam/T are counted by TMSCLEAN and included in the report totals on TMS Report 85. CA 1 will not scratch tapes owned by CA Dynam/T.

CA 1 must be active or batch active to run TMSCLEAN.

Report Description

TMSCLEAN generates TMS Report 85 to list general processing messages and any errors encountered. The parameters provided for TMSCLEAN are listed, and the CA 1 system options that affect the option are chosen. The data set name and volume serial number of the EARLOUT data set is also listed. Totals for each option are on TMS Report 85, and include the number of records read and updated, user exit statistics and other processing totals.

The supplied CA Earl report members can produce the following reports from the flat file defined by the EARLOUT DD:

CA Earl Member	TMSCLEAN Option	Report
TMECLN01	SCRATCHLIST	Scratch Listing
TMECLN02	RESET	Scratch and Clean Reset
TMECLN03	EXTEND	Extend Expiration Dates
TMECLN04	SCRATCHRELIST	Scratch and Clean Relist
TMECLN05	RESETCAT	Reset Catalog Control

A brief description of each report follows:

- Each tape marked for scratch is listed in scratch tape pool order and then by label type, density, recording technique and volume serial number order (as provided in the sample TMECLN01 member). The report lists volume sequence, data set name, tape type (CRTG/REEL), cleaning history number, date and usage, first volume, CA 1 keyword, FLAG1 (F1), FLAG2 (F2), FLAG3 (F3), FLAG4 (F4) and new out-of-area code.

  If TMSCLEAN is unable to determine the catalog status of a data set or experiences an error uncataloging a data set, the message LE=xx or UE=xx is printed for that data set on the output report, under the field heading UNCAT IND. The value of xx reflects the return code from the LOCATE or CAMLIST macro, respectively. To determine the cause of the LOCATE or CAMLIST error, see the *IBM MVS Catalog Administration Guide*. The TMC record is updated regardless of the return codes.

- The Scratch and Clean Relist report produced by the supplied TMECLN04 member lists all active tapes in scratch status in volume serial number order, and reports the date each tape had been scratched, the number of days the tape has been in scratch status, the last job used, the last date used, and whether the tape was marked for special handling when it was scratched.

  Note: If automatic cleaning is to take place, see TMSCLNOA (see page 175) in the following pages.

- The Extend Expiration Dates report produced by the supplied TMECLN03 member identifies data sets whose expiration dates have been extended through the EXTEND parameter. It lists the data sets in volume serial number order and gives the new expiration date, old expiration date, and the date last used.

- Scratch and Clean Reset report is produced by the TMECLN02 member. Any volume whose scratch bit has been reset is listed in volume serial number order with the expiration date, creation date and time, FLAG1 (F1), FLAG2 (F2), FLAG3 (F3) and FLAG4 (F4).
- The Reset Expired by Catalog Control report is produced by the TMECLN05 member. Any data sets that have previously been expired by Catalog Control are listed. TMC Volume and DSNB records marked as expired by Catalog Control are listed in data set order.

Scratch and Clean Listing

The Scratch and Clean Listing reports on all volumes that were scratched during the execution of TMSCLEAN with the SCRATCHLIST parameter and without the TEST parameter. A volume is considered eligible for scratch if the expiration date for File 1 is less than or equal to the execution date and the OUTCODE field contains hexadecimal zeros or a blank in the first position (but **not** (blank)LIB).

All multivolume pointers (1STVOL, NEXTVOL and PREVVOL) are cleared when volumes are scratched. Multi-data set pointers (1STDSNB, LSTDSNB and NUMDSNB) are cleared from the Volume record, and the DSNB records associated with the secondary files are cleared of information and returned to the *free chain* of available DSNBs.

Execution of various CA 1 utilities may be required prior to TMSCLEAN, depending on the CA 1 keyword dates in use.

- TMSEXPDT may be executed to reset the expiration criteria for eligible data sets. This utility is necessary only if you are using a Retention Data Set (RDS). Normally, TMSEXPDT overrides expiration dates only for data sets that used the CA 1 default (specified to the RP parameter in the TMOOPTxx member of *hlq*.CTAPOPTN). Optionally, JCL-supplied values can be overridden if the RO parameter in TMOOPTxx is set to YES or if the DEFAULT indicator was turned on when the TMC was initially formatted.
- TMSCTLG must be executed if you have data sets using Catalog Control on systems not yet running r5.2 or above. The program determines which Catalog Controlled data sets no longer have an entry in the MVS catalog and resets the expiration date to the current date plus any catalog extension (the number of days specified to the R9 parameter in the TMOOPTxx member of *hlq*.CTAPOPTN). Once all systems have been upgraded to r5.2 or higher and the TMSOSCAT utility has been run with the SYNC parameter, TMSCTLG no longer needs to be run.
- TMSCYCLE must be executed if you have data sets using Cycle Control. The program determines which data sets should be removed from Cycle Control, and resets the expiration date to the current date plus any cycle extension (the number of days specified to the RC parameter in the TMOOPTxx member of *hlq*.CTAPOPTN).

- TMSCLEAN may issue a return code four (RC=4). The possible causes are listed, refer to the report to diagnose the cause:

  1. SMS was unable to obtain MGMTCLAS definitions.
  2. SMS was unable to calculate the maximum retention period.
  3. Tape volume was not scratched due to chaining errors.
  4. Uncatalog or locate error occurred on a tape file.

Job Control Statements

```
//stepname EXEC PGM=TMSCLEAN,PARM='[TEST,]SCRATCHLIST[,F][,HOLDDAYS=n]'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TMSOPTNS DD  DSN=hlq.CTAPOPTN(TMSOPTNS),DISP=SHR
//EARLOUT  DD  DSN=scratch.list.earlout,                *NOTES
//             DISP=(NEW,CATLG,DELETE),
//             UNIT=SYSDA,
//             VOL=SER=VOLSER,
//             SPACE=(CYL,5),
//             DCB=RECFM=FB,LRECL=376,BLKSIZE=15416)    *NOTES
[//SCRACCT  DD  DSN=tape.acctng.info,UNIT=TAPE,DISP=(NEW,KEEP)      ]
[//             [,DCB=BLKSIZE=n*340]                     *NOTES    ]
[//TMSRPT89  DD SYSOUT=A]
[//SYSIN     DD *]
[VOL=ssssss-eeeeee]
[EXCLUDE=ssssss-eeeeee]
[SCRPOOL=xxxxx]
[FILE1DSN=pattern]
```

Consider the following:

- If the EARLOUT data set is used with TMSUNCAT to uncatalog data sets on other systems, catalog and save the data set. Use the EARLOUT data set as input to TMECLNxx to produce the reports for TMSCLEAN. The sample JCL provided has a step to execute CA Earl.
- The BLKSIZE can be any multiple of 376.
- The BLKSIZE can be any multiple of the LRECL, which is 340. If no BLKSIZE is specified and the release of the operating system supports system determined blocksize, the operating system selects the blocksize. Otherwise the BLKSIZE for SCRACCT and OUTAREA defaults to 32640 for tape or 8840 for DASD devices.

Parameter Definitions

The parameters must be specified in this order. All parameters except SCRATCHLIST are optional.

TEST

Process normally, but do not update any TMC records. The records processed are written to EARLOUT to subsequently produce the Scratch and Clean Listing, but the TMC is not updated.When the TEST parameter is used, the Scratch and Clean Listing does not include tapes expired from CATALOG, CATLG/ddd or LDATE/ddd control.

SCRATCHLIST

Indicates that tapes eligible for scratch are to be output to the EARLOUT data set, and subsequently used to produce the Scratch and Clean Listing.

F

The FAST paremeter can be used if tapes are under CATALOG control and the OCTLG option is set to NO in TMOOPTnn in CTAPOPTN. This parameter causes TMSCLEAN to stop processing on a DSNB chain as soon as it finds a DSNB that is still cataloged to the OS catalog. Without this parameter, TMSCLEAN processing will continue through the entire DSNB chain checking the OS catalog for all of them and expiring them if not found. This parameter is not recommended if you use CA 1 Copycat's FILECOPY with a control statement of FILES=ACTIVE. This will cause files to be copied that would have otherwise been expired.

HOLDDAYS=n

The HOLDDAYS parameter can be used if tapes are under CATALOG control and the OCTLG option is set to NO in TMOOPTnn in CTAPOPTN. This parameter indicates the minimum protection time of tape data sets under catalog control which were never cataloged. TMSCLEAN keeps these data sets n days before checking their catalog status, to protect long-running tape jobs. Valid values for n are 2 through 9. The default is HOLDDAYS=2.

JCL Considerations

SCRACCT DD

A TMC record image is written for every tape being scratched. You can use this information to determine the elapsed time of ownership for accounting and billing purposes (assuming CA 1 has been directed to capture accounting data). This DD statement is optional.

SYSIN DD

If the SYSIN DD is found, the TMC is processed based on the volumes that are specified in the range control statements. If no control statements are coded, the complete TMC is processed. A maximum of 20,000 range control statements is allowed. If the SYSIN DD is not found, then TMSCLEAN processes the complete TMC.

Example

```
//SYSIN DD *¨
VOL=ssssss-eeeeee¨
EXCLUDE=ssssss-eeeeee¨
SCRPOOL=xxxxx¨
```

TMSRPT89 DD

This DD must be coded, if the SYSIN DD is used. The range control statements are printed to this DD and any errors that found in the control statements are printed to this DD.

Range Control Statements

VOL=ssssss-eeeeee

The VOL statement is used to select the volumes to be processed based on a starting and ending volume number. You can specify a single volume by only coding a start volume.

Example: This control card processes volumes 001500-001524 and 001531-001550:

```
//SYSIN    DD  *
       VOL=001500-001550
       EXCLUDE=001525-001530
//
```

Note: Start and end volumes must be six digit numbers. If the volume number has five or fewer digits, it must be padded with 0 s, such as VOL=001500-001575.

SCRPOOL=xxxxx

The SCRPOOL statement is used to select the volumes to be processed based on a scratch subpool name.

FILE1DSN=pattern

The FILE1DSN statement is used to select the volumes based on the DSN of the first file on the volume, using standard CA 1 pattern masking. When the FILE1DSN statement is the only selection statement that is used, the complete TMC is processed and only volumes that match the specified pattern are processed. If FILE1DSN= statement is used with VOL and/or SCRPOOL statements, then only volumes within the requested ranges that match the pattern are processed.

EXCLUDE=ssssss-eeeeee

The EXCLUDE statement selects the volumes you want to exclude from processing based on a starting and ending volume number. To specify a single volume, code only a start volume.

Start and end volumes must be six digit numbers. If the volume number has five or fewer digits, it must be padded with 0s.

```
EXCLUDE=001500-001575
```

The EXCLUDE control statement can be used without a VOL control statement.

Example: The control statement excludes the volser range of 001500-001550 from being processed by TMSCLEAN.

```
EXCLUDE=001500-001550
```

Note: If the volumes included match the volumes excluded, the exclude control statement overrides the include control statement.

Scratch and Clean Relist

The SCRATCHRELIST parameter outputs Retention Records to the EARLOUT data set for all available scratch tapes. The EARLOUT data set can be used in a subsequent step to produce TMS Report 06 - Scratch and Clean Relist. All scratch tapes are listed regardless of when they became scratched.

The Scratch and Clean Relist, as provided in the TMECLN04 member, lists records by volume serial number and label type within each tape scratch pool. The type of media (CRTG for cartridge or REEL) is identified. Scratch tapes with a future expiration date (which is an abnormal condition) have an asterisk (*) in front of the volume serial number. CA 1 does not allow these tapes to be used as scratch tapes.

Job Control Statements

```
//stepname EXEC PGM=TMSCLEAN,PARM='[TEST,][SCRATCHRELIST]'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//EARLOUT  DD  DSN=scratch.list.earlout,
//             DISP=(NEW,PASS,DELETE),
//             UNIT=SYSDA,
//             SPACE=(CYL,5),
//             DCB=(RECFM=FB,LRECL=376,BLKSIZE=15416)      *NOTE
[//TMSRPT89    DD SYSOUT=A]
[//SYSIN       DD *]
[VOL=ssssss-eeeeee]
[EXCLUDE=ssssss-eeeeee]
[SCRPOOL=xxxxx]
[FILE1DSN=pattern]
```

Note: The BLKSIZE can be any multiple of 376.

Parameter Definitions

SCRATCHRELIST

Indicates that all available scratch tapes are to be output to the EARLOUT data set, and subsequently used to produce the Scratch and Clean Relist.

JCL Considerations

SYSIN DD

If the SYSIN DD is found, the TMC is processed based on the volumes specified in the range control statements. If no control statements are coded, the complete TMC is processed. A maximum of 20,000 range control statements is allowed. If the SYSIN DD is not found, then TMSCLEAN processes the complete TMC.

TMSRPT89 DD

If the SYSIN DD is used, this DD must be coded. The range control statements are printed to this DD and any errors that found in the control statements are printed to this DD.

Range Control Statements

VOL=ssssss-eeeeee

The VOL statement is used to select the volumes to be processed based on a starting and ending volume number. A single volume may be specified by only coding a start volume.

SCRPOOL=xxxxx

The SCRPOOL statement is used to select the volumes to be processed based on a scratch subpool name.

FILE1DSN=pattern

The FILE1DSN statement is used to select volumes based on the DSN of the first file on the volume, using standard CA 1 pattern masking. When the FILE1DSN statement is the only selection statement used, the complete TMC will be processed and only volumes that match the specified pattern are processed. If FILE1DSN= statement is used with VOL and/or SCRPOOL statements, then only volumes within the requested ranges that match the pattern are processed.

EXCLUDE=ssssss-eeeeee

The EXCLUDE statement selects the volumes you want to exclude from processing based on a starting and ending volume number. To specify a single volume, code only a start volume.

Start and end volumes must be six digit numbers. If the volume number has five or fewer digits, it must be padded with 0s.

```
EXCLUDE=001500-001575
```

The EXCLUDE control statement can be used without a VOL control statement.

Example: The control statement excludes the volser range of 001500-001550 from being processed by TMSCLEAN.

```
EXCLUDE=001500-001550
```

Note: If the volumes included match the volumes excluded, the exclude control statement overrides the include control statement.

Extend Expiration Dates

Another parameter associated with TMSCLEAN is EXTEND. This option extends expiration dates, and outputs Retention Records to the EARLOUT data set to subsequently produce TMS Report 27 - Extend Expiration Dates. This parameter can also extend the last used date by the number of days specified in the EXTEND parameter for volumes using the keyword, LDATE. This is done by defining the criteria in the EXTEND parameter. The ddd value in the LDATE/ddd keyword is updated.

CA Earl sample TMECLN03 can be used to print this report.

Job Control Statements

```
//stepname EXEC PGM=TMSCLEAN,
//             PARM='[TEST,]EXTEND[,LDATE],nn,xx,aaaaaa'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//EARLOUT  DD  DSN=extend.earlout,
//             DISP=(NEW,PASS,DELETE),
//             UNIT=SYSDA,
//             SPACE=(CYL,5),
//             DCB=(RECFM=FB,LRECL=376,BLKSIZE=15416)      *NOTE
[//TMSRPT89  DD SYSOUT=A]
[//SYSIN     DD *]
[VOL=ssssss-eeeeee]

[EXCLUDE=ssssss-eeeeee]
[SCRPOOL=xxxxx]
[FILE1DSN=pattern]
```

Note: The BLKSIZE can be any multiple of 376.

Parameter Definitions

TEST

Process normally, but do not update any TMC records. The records processed are written to EARLOUT to subsequently produce the Extend Expiration Dates report, but the TMC is not updated. This parameter is optional, and, if used, must be specified before EXTEND.

EXTEND

Extends the expiration dates on qualified TMC records.

LDATE

Allows LDATE/ddd expirations to be extended with all other qualified tapes.

nn

Is the number of days to extend the expiration date. If the tape has an expiration date less than the current date, it may still be eligible for scratch after adding the extension.

xx

Is the range of days from the current date within which the existing expiration date must fall to qualify for extension, that is, tapes expiring within xx days from the current date are to be extended nn days.

aaaaaa

Is the index of a data set or fully qualified data set name of data sets to be extended. The specification ALL may be used to extend all data sets, but when used with FILE1DSN control statements, a parm of ALL must be used.

JCL Considerations

SYSIN DD

If the SYSIN DD is found, the TMC is processed based on the volumes specified in the range control statements. If no control statements are coded, the complete TMC is processed. A maximum of 20,000 range control statements is allowed. If the SYSIN DD is not found, then TMSCLEAN processes the complete TMC.

TMSRPT89 DD

If the SYSIN DD is used, this DD must be coded. The range control statements are printed to this DD and any errors found in the control statements are printed to this DD.

Range Control Statements

VOL=ssssss-eeeeee

The VOL statement is used to select the volumes to be processed based on a starting and ending volume number. A single volume may be specified by only coding a start volume.

SCRPOOL=xxxxx

The SCRPOOL statement is used to select the volumes to be processed based on a scratch subpool name.

FILE1DSN=pattern

The FILE1DSN statement is used to select volumes based on the DSN of the first file on the volume, using standard CA 1 pattern masking. When the FILE1DSN statement is the only selection statement used, the complete TM will be processed and only volumes that match the specified pattern are processed. If FILE1DSN= statement is used with VOL and/or SCRPOOL statements, then only volumes within the requested ranges that match the pattern are processed.

EXCLUDE=ssssss-eeeeee

The EXCLUDE statement selects the volumes you want to exclude from processing based on a starting and ending volume number. To specify a single volume, code only a start volume.

Start and end volumes must be six digit numbers. If the volume number has five or fewer digits, it must be padded with 0s.

```
EXCLUDE=001500-001575
```

The EXCLUDE control statement can be used without a VOL control statement.

Example: The control statement excludes the volser range of 001500-001550 from being processed by TMSCLEAN.

```
EXCLUDE=001500-001550
```

Note: If the volumes included match the volumes excluded, the exclude control statement overrides the include control statement.

Extending Expiration Date Examples

1. An expiration date is extended four days for all data sets expiring during a four-day holiday weekend.

   ```
   //stepname EXEC PGM=TMSCLEAN,
   //             PARM='EXTEND,04,04,ALL'
   ```

2. An expiration date is extended four days for data sets beginning with the data set name ABC that are expiring during a four-day holiday period.

   ```
   //stepname EXEC PGM=TMSCLEAN,
   //             PARM='EXTEND,04,04,ABC'
   ```

3. All tapes including LDATE tapes that will expire during a four-day holiday weekend are extended.

   ```
   //stepname EXEC PGM=TMSCLEAN,
   //             PARM='EXTEND,LDATE,04,04,ALL'
   ```

Resetting the Scratch Bit for Future Expiration Dates

The RESET option processes all tapes with the scratch bit on and a future expiration date. It turns the scratch bit off and resets future clean dates to the current date. This option should *only* be used if the SCRATCHLIST option was executed after the system was IPLed with a future date. However, TMC record cleanup has already occurred, and other CA 1 utilities will have to be run to reestablish multivolume and multi-data set pointers and fields.

The RESET option will output Retention Records to the EARLOUT data set for subsequent report writer processing to produce TMS Report 42 - Scratch and Clean Reset. The volume serial number, volume sequence, data set name, expiration date, creation date and time, and other fields are listed on the report if the TMECLN02 member is used.

Job Control Statements

```
//stepname EXEC PGM=TMSCLEAN,
//             PARM='[TEST,]RESET'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//EARLOUT  DD  DSN=reset.earlout,
//             DISP=(NEW,PASS,DELETE),
//             UNIT=SYSDA,
//             SPACE=(CYL,5),
//             DCB=(RECFM=FB,LRECL=376,BLKSIZE=15416)       *NOTE
[//TMSRPT89  DD SYSOUT=A]
[//SYSIN     DD *]
[VOL=ssssss-eeeeee]
[EXCLUDE=ssssss-eeeeee]
[SCRPOOL=xxxxx]
[FILE1DSN=pattern]
```

Note: The BLKSIZE may be any multiple of 376.

Parameter Definitions

TEST

Process normally, but do not update any TMC records. The records processed are written to EARLOUT to subsequently produce the Scratch and Clean Reset report, but the TMC is not updated. This parameter is optional, and, if used, must be specified before RESET.

RESET

Turns off the scratch bit. (Records with the scratch bit on and a future expiration date).

JCL Considerations

SYSIN DD

If the SYSIN DD is found, the TMC is processed based on the volumes specified in the range control statements. If no control statements are coded, the complete TMC is processed. A maximum of 20,000 range control statements is allowed. If the SYSIN DD is not found, then TMSCLEAN processes the complete TMC.

TMSRPT89 DD

If the SYSIN DD is used, this DD must be coded. The range control statements are printed to this DD and any errors that found in the control statements are printed to this DD.

Range Control Statements

VOL=ssssss-eeeeee

The VOL statement is used to select the volumes to be processed based on a starting and ending volume number. A single volume may be specified by only coding a start volume.

SCRPOOL=xxxxx

The SCRPOOL statement is used to select the volumes to be processed based on a scratch subpool name.

FILE1DSN=pattern

The FILE1DSN statement is used to select volumes based on the DSN of the first file on the volume, using standard CA 1 pattern masking. When the FILE1DSN statement is the only selection statement used, the complete TMC will be processed and only volumes that match the specified pattern are processed. If FILE1DSN= statement is used with VOL and/or SCRPOOL statements, then only volumes within the requested ranges that match the pattern are processed.

EXCLUDE=ssssss-eeeeee

The EXCLUDE statement selects the volumes you want to exclude from processing based on a starting and ending volume number. To specify a single volume, code only a start volume.

Start and end volumes must be six digit numbers. If the volume number has five or fewer digits, it must be padded with 0s.

```
EXCLUDE=001500-001575
```

The EXCLUDE control statement can be used without a VOL control statement.

Example: The control statement excludes the volser range of 001500-001550 from being processed by TMSCLEAN.

```
EXCLUDE=001500-001550
```

Note: If the volumes included match the volumes excluded, the exclude control statement overrides the include control statement.

Resetting Data Sets Expired by Catalog Control

The RESETCAT option processes all data sets that have been expired by Catalog Control. Expired by Catalog Control is indicated in the TMC Volume Record when the X'08' bit is on in FLAG2. DSNB records are expired by Catalog Control when the X'20' is on in FLAG1.

This option should be used only when tapes are still cataloged, and have been expired by the utility TMSCTLG. This might occur if a system catalog containing tape data sets was not available or provided to TMSCTLG. The use of the RESETCAT option must be performed prior to executing TMSCLEAN with the SCRATCHLIST option, since multivolume and multifile chains are cleared during the SCRATCHLIST process.

The RESETCAT option produces the EARLOUT flat file, which can be used in a subsequent step (CA Earl TMECLN05) to produce a listing of TMC Volume and DSNB data sets that are processed by RESETCAT. Each data set will have its expiration date changed back to CATALOG, and the expired by Catalog Control indicator in the Volume and DSNB record is turned off.

Job Control Statements

```
//stepname EXEC PGM=TMSCLEAN,
//            PARM='[TEST,]RESETCAT'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//EARLOUT  DD  DSN=resetcat.earlout,
//            DISP=(NEW,PASS,DELETE),
//            UNIT=SYSDA,
//            SPACE=(CYL,5),
//            DCB=(RECFM=FB,LRECL=376,BLKSIZE=15416)       *NOTE
[//TMSRPT89  DD SYSOUT=A]
[//SYSIN    DD *]
[VOL=ssssss-eeeeee]
[EXCLUDE=ssssss-eeeeee]
[SCRPOOL=xxxxx]
[FILE1DSN=pattern]
```

Note: The BLKSIZE may be any multiple of 376.

Parameter Definitions

TEST

Process normally, but do not update any TMC records. The records processed are written to EARLOUT to subsequently produce the Reset Catalog Control report, but the TMC is not updated. This parameter is optional, and if used, must be specified before RESETCAT.

RESETCAT

Changes expiration date to CATALOG and turns off the Expired by Catalog Control indicator for TMC Volume and DSNB records if the Expired by Catalog Control indicator is on.

JCL Considerations

SYSIN DD

If the SYSIN DD is found, the TMC is processed based on the volumes specified in the range control statements. If no control statements are coded, the complete TMC is processed. A maximum of 20,000 range control statements is allowed. If the SYSIN DD is not found, then TMSCLEAN processes the complete TMC.

TMSRPT89 DD

If the SYSIN DD is used, this DD must be coded. The range control statements are printed to this DD and any errors that found in the control statements are printed to this DD.

Range Control Statements

VOL=ssssss-eeeeee

The VOL statement is used to select the volumes to be processed based on a starting and ending volume number. A single volume may be specified by only coding a start volume.

SCRPOOL=xxxxx

The SCRPOOL statement is used to select the volumes to be processed based on a scratch subpool name.

FILE1DSN=pattern

The FILE1DSN statement is used to select volumes based on the DSN of the first file on the volume, using standard CA 1 pattern masking. When the FILE1DSN statement is the only selection statement used, the specified pattern are processed. If FILE1DSN= statements is used with VOL and/or SCRPOOL statements, then only volumes within the requested ranges that match the pattern are processed.

EXCLUDE=ssssss-eeeeee

The EXCLUDE statement selects the volumes you want to exclude from processing based on a starting and ending volume number. To specify a single volume, code only a start volume.

Start and end volumes must be six digit numbers. If the volume number has five or fewer digits, it must be padded with 0s.

```
EXCLUDE=001500-001575
```

The EXCLUDE control statement can be used without a VOL control statement.

Example: The control statement excludes the volser range of 001500-001550 from being processed by TMSCLEAN.

```
EXCLUDE=001500-001550
```

Note: If the volumes included match the volumes excluded, the exclude control statement overrides the include control statement.

TMS Report 87 - Scratch and Clean Listing for SL CRTG Volumes

```
THURSDAY  , JUNE       09, 2005.158                      Y O U R   C O M P A N Y   N A M E                        PAGE      2

TMS REPORT-87        SCRATCH AND CLEAN LISTING FOR 'SL ' CRTG VOLUMES FOR SCRATCH SUBPOOL -
-----------------------------------------------------------------------------------------------------------------------------
VOLSER  FILE DATA SET NAME                                  CLN DATE     USE BASE   CA 1  SPEC. UNCATALOG F1 F2 F3 F4 TOTAL
         SEQ                                                CNT CLEANED  CLN VOLSER KEYWD HNDLE INDICATOR             ERRORS
-----------------------------------------------------------------------------------------------------------------------------
TDI010     1 CAI.CA1.TMS52.DDRTESTI                             ZEROS        TDI010 CATLG                 45 C0 04 08
TDI021     1 CAI.CA1.TMS52.TAPE02                               ZEROS        TDI021 CATLG                 E5 C0 00 08
TDI031     1 CAI.CA1.SRMDAL.DATAIN                              ZEROS        TDI031                       45 00 04 00
TDI049     1                                                    ZEROS        TDI049 CATLG *AIVS           E7 80 00 48 KEY2
980023     1 SYS4.TEST.FOREIGN1                                 ZEROS        980023 CATLG *AIVS           E6 80 00 48
980032     1 SYS4.TEST.SHOW                                     ZEROS        980032       *AIVS           E6 00 00 40

LABEL TYPE TOTAL
         SCRATCH TAPES      =    23
         MARKED FOR CLEANING=     1
         MARKED AS *BAD*    =     0
         NEED TO BE RE-INIT =     3
         NUMBER OF FOREIGN  =     0
         NEED TO BE ERASED  =     0


SCRATCH POOL TOTAL
         SCRATCH TAPES      =    24
         MARKED FOR CLEANING=     1
         MARKED AS *BAD*    =     0
         NEED TO BE RE-INIT =     3
         NUMBER OF FOREIGN  =     0
         NEED TO BE ERASED  =     0

MEDIA TYPE TOTAL
         SCRATCH TAPES      =    24
         MARKED FOR CLEANING=     1
         MARKED AS *BAD*    =     0
         NEED TO BE RE-INIT =     3
         NUMBER OF FOREIGN  =     0
         NEED TO BE ERASED  =     0


GRAND TOTAL
         SCRATCH TAPES      =    24
         MARKED FOR CLEANING=     1
         MARKED AS *BAD*    =     0
         NEED TO BE RE-INIT =     3
         NUMBER OF FOREIGN  =     0
         NEED TO BE ERASED  =     0
```

TMS Report 85 - TMSCLEAN Messages

```
TUESDAY, MAY 03, 2005.128                        Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-85      TMSCLEAN MESSAGES

PARAMETER LIST: TEST,SCRATCHLIST


Options in effect: TMCDSN  ASM.CA1DV.TMC2X
                   AUDDSN  ASM.CA1DV.AUDITX
                   UNCATA  N
                   SCRTCH  Y
                   DSEALL  N
                   UXCLN1  N
                   UXCLN2  N
                   UXSCR   N
                   SMS     Y
                   SR      N
                   OCTLG   N
                   SCRCAT  G
                   R9      0000000

Opening DDNAME SCRACCT: SYS96128.T164354.RA000.WITRU01B.TEMP
                        on volume TSO76A

Opening DDNAME EARLOUT: WITRU01.TEST.EARLOUT
                        on volume TSO76A
** WARNING VOLSER=980006 was not scratched => VOLSER IS NOT EXPIRED

                2,097 Records read

                    0 Records bypassed by user exit

                    0 Records modified by user exit

                    0 USER/NNN EXPDTs modified by user exit

                    0 USER/NNN EXPDTs expired  by user exit

                    0 CATLG/DDD EXPDTs changed to CATALOG

                   24 Tapes would have been scratched

                    1 Virutal WORM tapes scratched

                    0 Locates performed

                    0 Files would have been uncataloged

                   34 Records written to EARLOUT DD

                   24 Records written to SCRACCT DD
```

```
TUESDAY, MAY 03, 2005.128                        Y O U R   C O M P A N Y   N A M E                     PAGE=00002
TMS REPORT-85      TMSCLEAN MESSAGES
PARAMETER LIST: TEST,SCRATCHLIST


                  323 Previous scratched tapes unused
                   25 Tape sets scratched today
                   24 Tapes scratched today
                  347 Current scratch tapes
                1,644 Unexpired tape sets
                    2 Tapes not expired due to invalid last used date and EXPDT of LDATE/DDD
                    8 CATLG/DDD DSNs not processed due to invalid creation date

                    0 Tapes due to expire by MMMDD YYYY  (Tomorrow)
                    0 Tapes due to expire by MMMDD YYYY  ( 7 Days )
                    3 Tapes due to expire by MMMDD YYYY  (30 Days )
                   37 Tapes are using JULIAN expiration dates
                    2 Tapes are using LDATE control
                   18 Tapes are using CATALOG control
                    0 Tapes are using CYCLE control
                    0 Tapes are using PERMANENT retention control
```

```
TUESDAY, MAY 03, 2005.128                        Y O U R   C O M P A N Y   N A M E                                    PAGE=00003
TMS REPORT-85      TMSCLEAN MESSAGES
PARAMETER LIST: TEST,SCRATCHLIST
                    0 Tapes are using STATS control
                1,587 Tapes are using USER control
                    0 Tapes are controlled by an EDM
                    0 Tapes are 3592 WORM cartridges
                    0 Tapes are Virtual WORM cartridges
                    0 Tapes are managed by CA Dynam/T

                   12 Tapes held less than 8 days
                    0 Tapes held greater than 7 but less than 15 days
                    1 Tapes held greater than 14 but less than 31 days
                    0 Tapes held greater than 30 but less than 61 days
                    0 Tapes held greater than 60 but less than 91 days
                    0 Tapes held greater than 90 but less than 181 days
                    1 Tapes held greater than 180 but less than ONE year
                   11 Tapes held greater than ONE (1) year
```

TMS Report 06 - Scratch and Clean Relist

A TMSCLEAN SCRATCHRELIST operation produced the following sample report.

```
SATURDAY, FEBRUARY 26, 2005.055                      Y O U R   C O M P A N Y   N A M E                             PAGE      1

                                                 PARM=SCRATCHRELIST

TMS REPORT-06         SCRATCH AND CLEAN RELIST FOR 'SL ' CRTG VOLUMES FOR SCRATCH SUBPOOL - PATTPOOL

------------------------------------------------------------------------------------------------------------------------------
VOLSER    DATE           SCRATCH     LAST         LAST USE       CLEAN     CREATE       DATA SET NAME
          SCRATCHED         DAYS     JOB          DATE           TAPE      JOBNAME
------------------------------------------------------------------------------------------------------------------------------
980010    YYYY/001          2475     PRE-TMS      ZEROS                    PRE-TMS
980011    YYYY/001          2475     PRE-TMS      ZEROS                    PRE-TMS
980012    YYYY/001          2475     PRE-TMS      ZEROS                    PRE-TMS
980013    YYYY/056             1     JOBTAPER     YYYY/054                 JOBTAPER     CAI.CA1.MVMF.DATA
980014    YYYY/056             1     JOBTAPER     YYYY/054                 JOBTAPER     CAI.CA1.MVMF.DATA
980015    YYYY/056             1     JOBTAPER     YYYY/054                 JOBTAPER     CAI.CA1.MVMF.DATA

LABEL TYPE TOTAL
         SCRATCH TAPES      =     6
         MARKED FOR CLEANING=     0
         MARKED AS *BAD*    =     0


SCRATCH POOL TOTAL
         SCRATCH TAPES      =     6
         MARKED FOR CLEANING=     0
         MARKED AS *BAD*    =     0
```

TMS Report 42 - Scratch and Clean Reset

A TMSCLEAN RESET operation produced the following sample report.

```
SUNDAY, FEBRUARY 20, 2005.056                      Y O U R   C O M P A N Y   N A M E                               PAGE      1

                                                 PARM=RESETCAT

TMS REPORT-TMECLN05   RESET CATALOG

------------------------------------------------------------------------------------------------------------------------------
VOLSER       VOL    DATA SET NAME                             EXPIRATION    CREATE          CREATE     F1     F2     F3     F4
             SEQ                                              DATE          DATE              TIME
------------------------------------------------------------------------------------------------------------------------------
980006        1     CAI.CA1.MULTIVOL                          CATALOG       YYYY/034      13:09:00     61     40     00     00
980007        2     CAI.CA1.MULTIVOL                          CATALOG       YYYY/034      13:09:00     E1     40     00     00
980016        1     CAI.CA1.TAPER.OU01SL01                    CATALOG       YYYY/054      23:07:00     E1     40     02     00

END OF REPORT
```

Scratch and Clean RESETCAT

A TMSCLEAN RESETCAT operation produced the following sample report.

```
SUNDAY, FEBRUARY 20, 2005.056                      Y O U R   C O M P A N Y   N A M E                               PAGE      1

                                                 PARM=RESETCAT

TMS REPORT-TMECLN05   RESET CATALOG

------------------------------------------------------------------------------------------------------------------------------
VOLSER       VOL    DATA SET NAME                             EXPIRATION    CREATE          CREATE     F1     F2     F3     F4
             SEQ                                              DATE          DATE              TIME
------------------------------------------------------------------------------------------------------------------------------
980006        1     CAI.CA1.MULTIVOL                          CATALOG       YYYY/034      13:09:00     61     40     00     00
980007        2     CAI.CA1.MULTIVOL                          CATALOG       YYYY/034      13:09:00     E1     40     00     00
980016        1     CAI.CA1.TAPER.OU01SL01                    CATALOG       YYYY/054      23:07:00     E1     40     02     00

END OF REPORT
```

TMS Report 89 - Range Exception Report

A TMSCLEAN RANGE operation produced the following sample report.

```
THURSDAY, JUNE 09, 2008.158                        Y O U R   C O M P A N Y   N A M E                                 PAGE     1

TMS REPORT-89         TMSCLEAN RANGE REPORT

CONTROL STATEMENT IN EFFECT:
------------------------------------------------------------------------
VOL=A00000,A00005
VOL=CA3000 CA4000
SCRPOOL=SPOOL2
FILE1DSN=THIS.IS.*.TEST.FILE
FILE1DSN=THIS.IS.NOT.TEST.**
```

TMS Report 27 - Extend Expiration Dates

A TMSCLEAN EXTEND operation produced the following sample report.

```
SUNDAY, FEBRUARY 20, 2005.057                      Y O U R   C O M P A N Y   N A M E                             PAGE      1

                                                PARM=EXTEND,LDATE,04,07,ALL

TMS REPORT-27         EXTEND EXPIRATION DATES   4 DAYS

----------------------------------------------------------------------------------------------------------------------------
        VOLSER         VOL      DATA SET NAME                                 NEW            OLD            LAST USE
                       SEQ                                                    EXPDT          EXPDT          DATE
----------------------------------------------------------------------------------------------------------------------------
        980000          1       CAI.CA1.MVMF.OU01SL01                         LDATE/003      YYYY/055       YYYY/059
        980001          2       CAI.CA1.MVMF.OU01SL01                         LDATE/003      YYYY/055       YYYY/059
        980002          3       CAI.CA1.MVMF.OU01SL01                         LDATE/003      YYYY/055       YYYY/059
        980003          1       CAI.CA1.TAPER.TEST                            LDATE/003      YYYY/055       YYYY/059
        980005          1       CAI.CA1.DATA                                  YYYY/060       YYYY/056       YYYY/034
        980008          1       CAI.CA1.TEST                                  YYYY/057       YYYY/053       ZEROS
        980013          1       CAI.CA1.MVMF.DATA                             LDATE/008      YYYY/054       YYYY/058
        980014          2       CAI.CA1.MVMF.DATA                             LDATE/008      YYYY/054       YYYY/058
        980015          3       CAI.CA1.MVMF.DATA                             LDATE/008      YYYY/054       YYYY/058

END OF REPORT
```

TMSCLEAN Report Field Definitions

Options in effect: xxxxxxxx

CA 1 system options and values used by program

Opening DDNAME xxxxxxxx: On volume xxxxxxxx

Data set name and volume used by ddname listed

List of out of area control statements read

Control statement number and control statement image

Summary of tape sent out of area

Number of tape volumes sent out-of-area by out-of-area code and control statement number

Records read

Number of volume records read from TMC

Data sets expired by CATLG control reset

Number of data sets expired by TMSCTLG whose expiration dates have been reset to CATLG

Data sets expired by CATLG Control to be reset

Number of data sets expired by TMSCTLG whose expiration dates would have been reset if not TEST mode

Records bypassed by user exit

Number of volume records bypassed by TMSXCLN1 user exit

Records modified by user exit

Number of volume records modified by TMSXCLN1 user exit prior to TMC update

USER/NNN EXPDTs modified by user exit

Number of volume records using USER/uuu modified by TMSXCLN2 user exit

USER/NNN EXPDTs expired by user exit

Number of volume records using USER/nnn expired by TMSXCLN2 user exit

CATLG/DDD EXPDTs changed to CATALOG

Number of volume records whose expiration date have been reset from CATLG/ddd to CATALOG

Tapes would have been scratched

Number of volume records that would have been scratched if not TEST mode

Locates performed

Number of LOCATEs issued during processing

Files would have been uncataloged

Number of data sets that would have been uncataloged if not TEST mode

Records written to xxxxxxxx DD

Number of records written to ddname listed

Previous scratch tapes unused

Number of tape volumes in scratch status before processing

Tapes scratched today

Number of tape volumes scratched today

Virtual WORM tapes scratched

Number of virtual WORM tapes scratched

Tape sets scratched today

Number of single-volume and multi-volume sets that were scratched today.

Current scratch tapes

Number of tape volumes in scratch status after processing

Unexpired tape volumes

Number of tape volumes not in scratch status after processing

CATLG/DDD DSNs not processed due to invalid creation date

Number of tape volumes using CATLG/ddd not processed due to invalid creation date

Tapes due to expire by xxxxxxxxxx (Tomorrow)

Number of tape volumes estimated to scratch the next day

Tapes due to expire by xxxxxxxxxx (7 DAYS)

Number of tape volumes estimated to scratch by next week

Tapes due to expire by xxxxxxxxxx (30 DAYS)

Number of tape volumes estimated to scratch by next month

Tapes not expired due to invalid last used date and EXPDT of LDATE/DDD

Number of tape volumes using LDATE/ddd not expired due to invalid last used date

Tapes are using JULIAN expiration dates

Number of tape volumes not using CA 1 keyword retention control

Tapes are using LDATE control

Number of tape volumes using Days Since Last Used Retention Control

Tapes are using CATALOG control

Number of tape volumes using Catalog Retention Control and Catalog Days Retention Control

Tapes are using CYCLE control

Number of tape volumes using Cycle Retention Control

Tapes are using PERMANENT retention control

Number of tape volumes using Permanent Retention Control, excluding EDM

Tapes are using STATS control

Number of tape volumes using Status Retention Control

Tapes are using USER control

Number of tape volumes using User Retention Control

Tapes are controlled by an EDM

Number of tape volumes managed by an External Data Manager

Tapes are 3592 WORM cartridges

Number of 3592 WORM tapes protected by media type

Tapes are Virtual WORM cartridges

Number of virtual WORM cartridges found during this run

Tapes are managed by CA Dynam/T

Number of tape volumes managed by CA Dynam/T

Tapes held less than 8 days

Number of tape volumes scratched today 0-7 days from creation date

Tapes held greater than 7 but less than 15 days

Number of tape volumes scratched today 8-14 days from creation date

Tapes held greater than 14 but less than 31 days

Number of tape volumes scratched today 15-30 days from creation date

Tapes held greater than 30 .but less than 61 days

Number of tape volumes scratched today 31-60 days from creation date

Tapes held greater than 60 but less than 91 days

Number of tape volumes scratched today 61-90 days from creation date

Tapes held greater than 90 but less than 181 days

Number of tape volumes scratched today 91-180 days from creation date

Tapes held greater than 180 but less than ONE year

Number of tape volumes scratched today 181-365 days from creation date

Tapes held greater than ONE (1) year

Number of tape volumes scratched today more than one year from creation date

Tapes have EXPDTS that have been reset

Number of tape volumes with expirations that have been reset

Tapes have EXPDTS that would have been reset

Number of tape volumes with expirations that would have been reset if not TEST mode

Tapes have EXPDTS that have been EXTENDED

Number of tape volumes with expirations that have been extended

Tapes have EXPDTS that would have been EXTENDED

Number of tape volumes with expirations that would have been extended if not TEST mode

PARAMETER LIST

Parameters passed to program

VOLSER

Volume serial number of tape volume

FILE SEQ

File sequence number of file

DATA SET NAME

Data set name of tape volume or file

CLN CNT

Number of cleanings for tape volume since initialization

DATE CLEANED

Last date tape volume cleaned

USE CLN

Number of uses since last clean date for tape volume

BASE VOLSER

First tape volume of multivolume chain

CA 1 KEYWD

Tape volume expired by CA 1 keyword indicator

SPEC HNDLE

Special handling indicator, may contain *BAD*, CLEAN, ERASE, *AIVS, or *FORN

UNCATALOG INDICATOR

Uncatalog tape data set indicator where the following would indicate the status of the uncataloging:

UNCAT

Tape data set was successfully uncataloged

LE=rc

There was a LOCATE error, rc is the return code from the LOCATE macro

UE=rc

There was an UNCATALOG error, rc is the return code from the CAMLIST macro

blank

Tape data set not cataloged

F1

FLAG1 value of tape volume or file

F2

FLAG2 value of tape volume or file

F3

FLAG3 value of tape volume

F4

FLAG4 value of tape volume

TOTAL ERRORS

Total permanent and temporary Read and Write errors (TMECLN01)

SCRATCH TAPES

Number of tape volumes scratched today for this group

MARKED FOR CLEANING

Number of tape volumes flagged for cleaning today for this group

MARKED AS *BAD*

Number of tape volumes flagged for certification today for this group

NEED TO BE RE-INIT

Number of tape volumes that were AIVS volumes and need to be reinitialized prior to reuse

NUMBER OF FOREIGN

Number of tape volumes that were foreign volumes and need to be returned to their owner

NUMBER OF ERASED

Number of tape volumes that contained sensitive data and need to be erased or degaussed prior to reuse

CONTROL STATEMENT IN EFFECT

Range control statements processed

INVALID CONTROL STATEMENT

Invalid range control statement was read, valid control statements are:VOL= , SCRPOOL=, and FILE1DSN=

INVALID START VOLUME, VOLUME NOT IN TMC

The volume uses as the start of a volume range is not found in the TMC

INVALID END VOLUME, VOLUME NOT IN TMC

The volume uses as the end of a volume range is not found in the TMC

INVALID RANGE, START VOLUME GREATER THEN END VOLUME

The volume uses as the start of a volume range is greater then the volume used as the end of a range

NO SCRATCH SUBPOOLS USED BY CA 1

No SCRATCH SUBPOOLS are defined to CA 1, the SCRPOOL= control statement can not be used

SCRATCH SUBPOOL NOT DEFINED TO CA 1

The SCRATCH SUBPOOLS name is not defined to CA 1

TMSCLNOA Utility—Clean and Mark Out-of-Area

Use the TMSCLNOA utility to mark volumes that need to be cleaned and sent off-site.

This is an optional step that can be run after TMSCLEAN, and before the CA Earl reports. This 3592 WORM cartridges cannot be processed by TMSCLNOA, because they are not scratched by TMSCLEAN. Sometimes it may be convenient to send tapes to an out-of-area location as they are marked scratch (for example, a date entry system located on a non-MVS controlled system). Once marked out-of-area by TMSCLNOA, these tapes are automatically checked in when used for input on the CA 1 controlled system.

It may be desired to control which volumes need to be cleaned based on:

- the number of uses since the last cleaning
- the number of days since the last cleaning
- a target value has been reached for any of the 4 error conditions since the last cleaning which include:
  - temporary write errors
  - temporary read errors
  - permanent write errors
  - permanent read errors

With the control statements, you may AND together or OR together any combination of cleaning criteria.

CA 1 must be active or batch-active to run TMSCLNOA to update the volume records. If TMSCLNOA is run with a PARM=TEST, CA 1 may be inactive, batch-active or active.

Report Description

TMSCLNOA produces TMS Report 86 - TMSCLNOA Messages. A sample of this report showing some of the messages appears on page 4 (see page 179). Errors are listed for:

- any control statements in error
- total number of volumes that were updated (or would have been updated if a PARM=TEST was used)
- the number of records read from the EARLIN file
- the number of records written out to the optional OUTAREA file

The supplied CA Earl report member TMECLNOA produces the Scratch and Clean Listing (TMS Report 87) from the flat file defined by the EARLOUT DD created by TMSCLNOA. For sample JCL see SAMPLIB member JOBCLNOA

Job Control Statements

```
   //TMSCLNOA EXEC PGM=TMSCLNOA [,PARM=TEST]
   //STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
   //SYSUDUMP DD  SYSOUT=*
   //TMSRPT   DD  SYSOUT=*
[  //OUTAREA  DD  DSN=tapes.sent.out.of.area,UNIT=TAPE,         ]
[  //  DISP=(,KEEP)  (,DCB=BLKSIZE=n*340)                       ]
   //EARLIN   DD  DSN=earlout.from.tmsclean,DISP=(OLD,PASS)
   //EARLOUT  DD  DSN=earlout.to.earl.report.writer,
   //    DISP=(,CATLG),UNIT=SYSDA,SPACE=(CYL,5),
   //    DCB=(RECFM=FB,LRECL=376,BLKSIZE=15416)
   //SYSIN    DD  *

   (control statements here)
   /*
```

Parameter Definitions

TEST

Process normally, but do not update any TMC records. All reports are produced and the EARLOUT is updated as if the TMC had been updated. This parameter is optional.

Control Statement Specification

There are two basic types of control statements; OUTCODE or CLEAN. Each type of control statement has different optional keywords that can be included.

OUTCODE Control Statement

Up to nine valid OUTCODE control statements may be included in a single execution of TMSCLNOA. If the OUTAREA DD is supplied, each TMC volume record selected to be checked out of the area is written to the OUTAREA DD. At least one additional keyword must be included with the OUTCODE keyword. The format for the OUTCODE control statement is;

```
OUTCODE=aaaa (,1ST=nnnnnn) (,VOLSERS=vvvvvv-vvvvvv) (,LBLTYP=typ) (,DEN=ddd)
```

Control Statement Definition

aaaa

Designates that if the tapes meet the specified criteria, they are checked out to the out-of-area code aaaa. The OUTCODE must be four characters, and the statement must begin in column one.

1ST

Specifies that once tapes meet any label type (LBLTYP) or density (DEN) requirements, the first nnnnnn scratch tapes are checked out-of-area to the OUTCODE specified. This keyword is optional but is mutually exclusive with the VOLSERS keyword.

VOLSERS

Specifies that all volume serial numbers within this range should be checked out-of-area to the specified OUTCODE, providing they meet the LBLTYP or DEN requirements. This keyword is optional but is mutually exclusive with the 1ST keyword.

LBLTYP

Specifies the label type requirement, where type is SL, NL, NSL, BLP, SUL, LTM, AL1, AL#, AU1 or AU3. If LBLTYP is not coded, any label type is acceptable. This keyword is optional.

DEN

Specifies the density requirement, where dddd is 200, 556, 800, 1600, 6250, 38K, 38KC, or 38X2. If DEN is not coded, any density is acceptable. This keyword is optional.

CLEAN Control Statement

Up to nine valid CLEAN control statements may be included in a single execution of TMSCLNOA. At least one additional keyword must be included with the CLEAN keyword. If multiple criteria keywords are included on a single statement, they must **all** be matched for the volume to be marked CLEAN. If multiple CLEAN control statements are used, and all the criteria keywords on any one of the CLEAN control statements are matched, the volume is marked CLEAN. The format for the CLEAN control statement is;

```
CLEAN (,DAYS=ddd) (,USES=nnn) (,TR=eeeee) (,PR=eeeee) (,TW=eeeee) (,PW=eeeee)
```

Control Statement Definition

DAYS

Specifies the number of days that must have passed since the last cleaning. The keyword is optional and if present must be 3 numeric digits.

USES

Specifies the number of uses since the last time the volume was cleaned. The keyword is optional and if present must be 3 numeric digits.

TR / PR / TW / PW

Specifies the number of Temporary Read errors, Permanent Read errors, Temporary Write errors, or Permanent Write errors that must have occurred since the last cleaning. These keywords are all optional, and if present must be 5 numeric digits.

Control Statement Example

```
col 1
1   CLEAN,PR=00005,PW=00005
2   OUTCODE=KEY1,1ST=15,LBLTYP=SL,DEN=6250
3   CLEAN,USES=100
4   OUTCODE=KEY2,VOLSERS=TDI040-TDI049
5   CLEAN,DAYS=180,TW=00100,TR=00100
```

1. Indicates that any volume with more than 5 Permanent Read errors and more than 5 Permanent Write errors should be marked for cleaning.
2. Indicates that the first 15 volumes with a label-type of SL and a density of 6250 should be marked out-of-area to location KEY1.
3. Indicates that any volume that has been used more than 100 times since the last cleaning should be marked for cleaning.
4. Indicates that all volumes between TDI040 and TDI049 that go scratch should be marked out-of-area to location KEY2.
5. Indicates that any volume that has not been cleaned for 180 days and has more than 100 Temporary Write errors and has more than 100 Temporary Read errors should be marked for cleaning.

TMS Report 86 - TMSCLNOA Messages

```
THURSDAY, MAY 22, 2005.144                       Y O U R   C O M P A N Y   N A M E                                    ROxxxxx PAGE=00001
TMS REPORT-86     TMSCLNOA MESSAGES
PARAMETER LIST: TEST


OPENING DDNAME EARLOUT: CAI.CA1.TEST.EARLOUT
                        On volume TS076A
Opening DDNAME OUTAREA: SYS96144.T172202.RA000.JOBCLNOA.TEMP
                        On volume ASM001
           List of out of area and clean control statements read
                                      1 CLEAN,USES=010,DAYS=180,PR=00100,TR=00200,PW=00300,TW=00400
                                      2 OUTCODE=KEY1,1ST=15,LBLTYP=SL,DEN=6250
                                      3 CLEAN,USES=002
                                      4 OUTCODE=KEY2,VOLSERS=TDI040-TDI049
                                      5 CLEAN,DAYS=365,PR=00010,PW=00001,TW=00005
        34 records have been read from EARLIN
         2 records would have been updated
         1 record written to OUTAREA file
```

TMS Report 05 - Scratch and Clean Listing for SL Tape Volumes

```
THURSDAY  , JUNE       02, 2005.158                      Y O U R   C O M P A N Y   N A M E                        PAGE      2

TMS REPORT-05        SCRATCH AND CLEAN LISTING FOR 'SL ' CRTG VOLUMES FOR SCRATCH SUBPOOL -
------------------------------------------------------------------------------------------------------------------------------
VOLSER  FILE DATA SET NAME                                  CLN DATE         USE BASE   CA 1  SPEC. UNCATALOG F1 F2 F3 F4  LOC
         SEQ                                                CNT CLEANED      CLN VOLSER KEYWD HNDLE INDICATOR               ID
------------------------------------------------------------------------------------------------------------------------------
TDI010     1 CAI.CA1.TMS52.DDRTESTI                             ZEROS            TDI010 CATLG                 45 C0 04 08
TDI021     1 CAI.CA1.TMS52.TAPE02                               ZEROS            TDI021 CATLG                 E5 C0 00 08
TDI031     1 CAI.CA1.SRMDAL.DATAIN                              ZEROS            TDI031                       45 00 04 00
TDI049     1                                                    ZEROS            TDI049 CATLG *AIVS           E7 80 00 48 KEY2
980023     1 SYS4.TEST.FOREIGN1                                 ZEROS            980023 CATLG *AIVS           E6 80 00 48
980032     1 SYS4.TEST.SHOW                                     ZEROS            980032       *AIVS           E6 00 00 40

LABEL TYPE TOTAL
         SCRATCH TAPES      =    23
         MARKED FOR CLEANING=     1
         MARKED AS *BAD*    =     0
         NEED TO BE RE-INIT =     3
         NUMBER OF FOREIGN  =     0
         NEED TO BE ERASED  =     0


SCRATCH POOL TOTAL
         SCRATCH TAPES      =    24
         MARKED FOR CLEANING=     1
         MARKED AS *BAD*    =     0
         NEED TO BE RE-INIT =     3
         NUMBER OF FOREIGN  =     0
         NEED TO BE ERASED  =     0

MEDIA TYPE TOTAL
         SCRATCH TAPES      =    24
         MARKED FOR CLEANING=     1
         MARKED AS *BAD*    =     0
         NEED TO BE RE-INIT =     3
         NUMBER OF FOREIGN  =     0
         NEED TO BE ERASED  =     0


GRAND TOTAL
         SCRATCH TAPES      =    24
         MARKED FOR CLEANING=     1
         MARKED AS *BAD*    =     0
         NEED TO BE RE-INIT =     3
         NUMBER OF FOREIGN  =     0
         NEED TO BE ERASED  =     0
```

TMSCLNOA Report Field Definitions

PARAMETER LIST

Parameters passed to program

VOLSER

Volume serial number of tape volume

FILE SEQ

File sequence number of file

DATA SET NAME

Data set name of tape volume or file

CLN CNT

Number of cleanings for tape volume since initialization

DATE CLEANED

Last date tape volume cleaned

USE CLN

Number of uses since last clean date for tape volume

BASE VOLSER

First tape volume of multivolume chain

CA 1 KEYWD

Tape volume expired by CA 1 keyword indicator

SPEC HNDLE

Special handling indicator, may contain *BAD*, CLEAN, ERASE, *AIVS, or *FORN

UNCATALOG INDICATOR

Uncatalog tape data set indicator where the following would indicate the status of the uncataloging:

UNCAT

Tape data set was successfully uncataloged

LE=rc

There was a LOCATE error, rc is the return code from the LOCATE macro

UE=rc

There was an UNCATALOG error, rc is the return code from the CAMLIST macro

F1

FLAG1 value of tape volume or file

F2

FLAG2 value of tape volume or file

F3

FLAG3 value of tape volume

F4

FLAG4 value of tape volume

LOC ID

Designates new out-of-area code of tape volume (TMECLNOA)

SCRATCH TAPES

Number of tape volumes scratched today for this group

MARKED FOR CLEANING

Number of tape volumes flagged for cleaning today for this group

MARKED AS *BAD*

Number of tape volumes flagged for certification today for this group

NEED TO BE RE-INIT

Number of tape volumes that were AIVS volumes and need to be reinitialized prior to reuse

NUMBER OF FOREIGN

Number of tape volumes that were foreign volumes and need to be returned to their owner

NUMBER OF ERASED

Number of tape volumes that contained sensitive data and need to be erased or degaussed prior to reuse

TMSCONVR Utility—Convert Sequential Data Sets

Use the TMSCONVR batch utility program to reformat any machine-readable data set into one of three possible formats.

Depending on the format you select, the output record formats are TMSUPDTE control statements, TMSEXPDT Retention Data Set (RDS) control statements, or TMRECORD (TMC record) format.

TMSCONVR performs several functions:

- SMF data collected from a non-CA 1 controlled computer can be used as input to TMSCONVR to generate TMSUPDTE control statements. These control statements can then be input to TMSUPDTE to update the TMC.
- TMSCONVR can reformat data from more than one source to use as input to TMSFORMT when initially creating the TMC.
- TMSCONVR can reformat input data for the Generalized Report Writer (TMSGRW).
- TMSCONVR can create the RDS used by TMSEXPDT.

A user exit, TMSXITCO, may be called from TMSCONVR to modify data in the output records.

CA 1 may be active, batch active or inactive to run TMSCONVR.

Report Description

Depending on the output record format you select, TMSCONVR generates one of the following reports:

- TMS Report 18 - User-Supplied Input Converted to TMRECORD Format (see page 200)
- TMS Report 18 - User-Supplied Input Converted to TMSEXPDT Format (see page 201)
- TMS Report 18 - User-Supplied Input Converted to TMSUPDTE Format (see page 201)

Input Data Sets

TMSCONVR can be executed with either two or three input files: the SYSIN and TMINPUT files or the SYSIN, ACTIVITY, and DEFAULTS files. The JCL parameters and the input files used determine the type of processing performed. When the TMINPUT data set is used, TMSCONVR reformats the data in TMINPUT into the proper output format: TMSUPDTE, TMSEXPDT or TMRECORD.

When the ACTIVITY and DEFAULTS data sets are used, TMSCONVR attempts to match a DEFAULTS record (a normal, basic, or skeleton record) with an ACTIVITY record (an override or transaction record) to produce a single output record properly formatted and containing data from both input records. The DEFAULTS data set contains the default values for data items. The ACTIVITY data set contains data that either overrides what is in the DEFAULTS data set or applies to fields not defined in the DEFAULTS data set.

TMINPUT and DEFAULTS must contain positional or fixed field data. Data for a specific field is in a user-defined position in the record. For example, the volume serial number might be in positions 13 through 18. The position of the data in TMINPUT or DEFAULTS is defined to TMSCONVR through the SYSIN data set. ACTIVITY must contain keyword data. The data is coupled with a field name (VOLSER=012345).

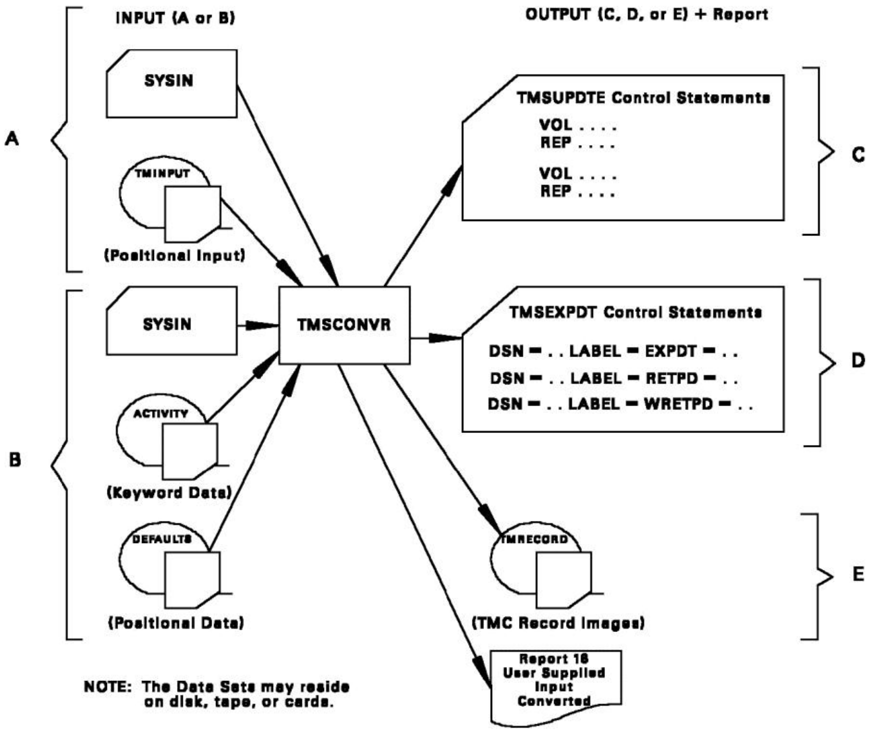
Input Restrictions for Date Fields

Date fields input for conversion are only accepted in packed decimal or character format. Date fields input in binary or hexadecimal format generates an error message and result in a user abend.

- If the input date fields are packed decimal, they **must** be in the format yyddd or ccyyddd. Any other format generates an error message and results in a user abend.
- Character dates may be input in any format compatible with the TMMDATE macro process. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

Date fields with null data contain all blanks, hexadecimal zeros or packed zeros. Except for records converted to TMSEXPDT output format, these fields are generally ignored during TMSCONVR processing. Any null EXPDT or RETPD generates an error message and a user abend.

The following diagram illustrates TMSCONVR Program Input and Output:



TMINPUT Data Set

The TMINPUT data set is a stand-alone file containing all the necessary data to build the output record requested. The data in this file must be positional. The fields that are to be used in the file **must** be described to TMSCONVR through the SYSIN data set.

Field and data are bypassed if the field contains only blanks, binary zeros or packed zeros.

Duplicate records are not allowed. A record is considered a duplicate if the following fields are equivalent to corresponding fields in another record:

- VOLSER (volume serial number) for TMSUPDTE output format and TMRECORD output format.
- DSN (data set name) and CJOB (creating job name) combination for TMSEXPDT output format. If the data set names are not qualified by a creating job name, duplicate data set names are not allowed.

If more than one physical record makes up one logical record, this **must** be indicated to TMSCONVR by use of the TMINPUT= parameter.

ACTIVITY Data Set

Records in the ACTIVITY data set contain keywords and values, such as VOLSER=345678,DSN=JKL,CJOB=ELMER. Valid keywords are listed on Keywords and Their Use (see page 199).

The keyword value must be in character format, and may only contain the characters A-Z, 0-9, $, # or @. Hexadecimal, packed and binary data are not allowed. If the value contains any special characters or embedded blanks, the value must be enclosed in delimiters. The delimiters must be asterisks (*) and cannot be embedded within the value.

Example:

```
ACCT=*CA-1 PROJECT*
```

The delimiters between keyword=value phrases may be commas, blanks, or multiple combinations of both. For example:

```
keyword=value,keyword=value keyword=value
```

The value HEXZEROS indicates binary zeros. When used, it clears the field.

Duplicate records are not allowed. A record is considered a duplicate if the following fields are equivalent to corresponding fields in another record:

- DSN (data set name) and VOLSER (volume serial number) for TMSUPDTE output format.
- DSN and CJOB (creating job name) for TMSEXPDT output format.
- DSN and VOLSER for TMRECORD output format.

The ACTIVITY data set **must** be used in conjunction with the DEFAULTS data set.

If more than one physical record makes up one logical record, this **must** be indicated to TMSCONVR by use of the MASTER= parameter.

DEFAULTS Data Set

The data in the DEFAULTS data set is positional. The location of the data must be defined to TMSCONVR using SYSIN control statements. The format of the SYSIN control statements is defined on n (see page 198).

Each record in the ACTIVITY data set is matched by data set name with a record in the DEFAULTS data set. For each record matched, one output record (or a group of output records, as in the case of TMSUPDTE output) is generated. Data in the ACTIVITY data set overrides the corresponding data in the DEFAULTS data set when the two matched records are merged for output. The DEFAULTS data (defined by the SYSIN control statements) that is not specified on the ACTIVITY data set is moved into the output record.

If there is a data set name in the ACTIVITY data set that is not matched with a DEFAULTS data set record, or vice versa, no output record is generated for that data set name. There **must** be a matching data set name for a record to be output. Field and data are bypassed when the field contains only blanks, binary zeros or packed zeros.

The DEFAULTS data set must not contain duplicate records. If more than one physical record makes up one logical record, this **must** be indicated to TMSCONVR by use of the DEFAULTS= parameter.

DEFAULTS Data

DSN	EXPDT	CJOB	CDATE
ABC		JOB1	341/YYYY
DEF	360/YYYY	JOB2	342/YYYY
JKL	324/YYYY	JOB3	343/YYYY
MNO	120/YYYY	JOB4	344/YYYY

ACTIVITY Data

DSN=ABC,VOLSER=012345,EXPDT=99000
DSN=DEF,VOLSER=123456
DSN=GHI,VOLSER=234567,CJOB=MAKE1
VOLSER=345678,DSN=JKL,CJOB=MAKE2

SYSIN Data

DSN=(1,5,CH)
EXPDT=(10,10,CH,DATEFMT=(DDD/YYYY))
CJOB=(20,8,CH)
CDATE=(30,10,CH,DATEFMT=(DDD/YYYY))

Output Records for TMSUPDTE

VOL	012345,NODSN
REP	DSN=*ABC*
REP	CDATE=*MMMDD YYYY*
REP	EXPDT=*CATALOG*
REP	CJOB=*JOB1*
VOL	123456,NODSN
REP	DSN=*DEF*
REP	CDATE=*MMMDD YYYY*
REP	EXPDT=*MMMDD YYYY*
REP	CJOB=*JOB2*
VOL	345678,NODSN
REP	DSN=*JKL*
REP	CDATE=*MMMDD YYYY*
REP	EXPDT=*MMMDD YYYY*
REP	CJOB=*MAKE2*

Note: There are no TMSUPDTE output records created for data sets MNO (DEFAULTS input) and GHI (ACTIVITY input). For an output record to be created, there must be a matching data set name for the DEFAULTS data set and the ACTIVITY data set.

Job Control Statements

Due to the number of options available with TMSCONVR, the JCL is presented separately for the three output formats available, TMRECORD, TMSEXPDT and TMSUPDTE.

Creating TMRECORD Output

Either the TMINPUT data set or the ACTIVITY and DEFAULTS data sets can be input to TMSCONVR to produce TMC record images. These records can then be input to TMSFORMT or TMSGRW.

The minimum data required as input to TMSCONVR is the data set name and volume serial number.

```
 //stepname EXEC PGM=TMSCONVR,
 //             PARM='parm'
 //STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
 //TMOUTPUT DD  DSN=tmrecord.images,
 //             DISP=(NEW,CATLG,DELETE),
 //             UNIT=SYSDA,SPACE=(space1),
 //             [DCB=BLKSIZE=N*340]                            *NOTE
 //SORTOUT  DD  UNIT=SYSDA,SPACE=(space2)
 //TMSRPT   DD  SYSOUT=A
 //SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
 //SYSOUT   DD  SYSOUT=A
 //SORTWK01 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK02 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK03 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK04 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK05 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK06 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //TMINPUT  DD  DSN=input,DISP=OLD¬
[//ACTIVITY DD  DSN=input,DISP=OLD   ]
[//DEFAULTS DD  DSN=defaults,DISP=OLD]
 //SYSWK01  DD  UNIT=SYSDA,SPACE=(space2)
 //SYSIN    DD  *
 (control statements here)
 /*
```

Note: The BLKSIZE can be any multiple of the LRECL, 340. If no BLKSIZE is specified, the BLKSIZE defaults to 6800.

Parameter Definitions

The following parameters may be specified in any order in the PARM= field. Each parameter may be specified only once.

DATEFMT=(fmt)

Specifies the preferred *output* date format for date fields being converted. If no value is specified, DATEFMT=(DEFAULT) is used. This date format is also used as the SYSIN *input* date format, unless a different date format is specified in SYSIN. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

TMRECORD

Identifies the output record format. This parameter is *required*.

UXCO=exit_name1

Identifies the user exit load module name that TMSCONVR uses. The UXCO module can be named anything. The name can be a maximum of eight bytes and must follow MVS naming conventions. This parameter is optional.

XITU=exit_name2

When using Alpha Numeric exits TMSXITU and TMSXITE, code the XITU exit name as a parameter. The name can be a maximum of eight bytes and must follow MVS naming conventions. This parameter is optional.

TMINPUT=[*1*|nn]

Relates to the TMINPUT data set. Cannot be used with the DEFAULTS=nn and/or ACTIVITY=nn parameters. The value of nn specifies the number of physical records that make one logical record. The default is 1. This parameter is optional.

MASTER=[*1*|nn]

Relates to the ACTIVITY data set. Cannot be used with the TMINPUT=nn parameter. The value of nn specifies the number of physical records that make one logical record. The default is 1. This parameter is optional.

DEFAULTS=[*1*|nn]

Relates to the DEFAULTS data set. Cannot be used with the TMINPUT=nn parameter. The value of nn specifies the number of physical records that make one logical record. The default is 1. This parameter is optional.

MAXERR=[*50*|nnnnn]

Specifies the number of errors accepted before the job takes a U0006 abend. The default is 50 and the maximum is 99999. This parameter is optional.

HEXZEROS

Causes the default values to be placed into fields not specified on the input file(s). See Keywords and Their Use (see page 199). Optional.

JCL Considerations

space1

Is the amount of space required for the number of records to be generated. The output record block size is 3400 (340-byte records blocked by 10).

space2

Is the amount of space required for records 386-bytes long, blocked 10 records per block. There is approximately one output record for each logical input record.

TMINPUT or ACTIVITY DEFAULTS

The input data set may reside on any input device as a sequential data set. If it is a member of a PDS (partitioned data set), it must be called as DSN=dsname (member) in the JCL. If it is on NL tapes, the DCB data must be provided. If it is statement input, it must be called as *,DCB=BLKSIZE=80 in the JCL.

An example PARM list is:

```
PARM='DATEFMT=(MMMDD YYYY),TMRECORD,MAXERR=99,HEXZEROS'
```

TMSCONVR creates output records in TMRECORD image format. All output dates are formatted according to the date pattern MMMDD YYYY. A maximum of 99 errors is accepted before the job terminates with a U006 abend. Values are assigned to fields not specified on the input files.

Creating TMSEXPDT Output

Either the TMINPUT data set or the ACTIVITY and DEFAULTS data sets can be input to TMSCONVR to produce TMSEXPDT (RDS) control statements. Omit unused DD statements. DD DUMMY is an active file.

The minimum data required as input is the data set name. If PARM=HEXZEROS is not specified (causing a default expiration date), the minimum data required is the data set name and an expiration date or retention period (RETPD or WRETPD). To generate a RETPD value for an RDS control statement, a value less than 10,000 must be passed from the input.

```
 //stepname EXEC PGM=TMSCONVR,
 //             PARM='parm'
 //STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
 //TMOUTPUT DD  DSN=tmsexpdt.control.statements,
 //             DISP=(NEW,CATLG,DELETE),
 //             UNIT=SYSDA,SPACE=(space1),
 //             [DCB=BLKSIZE=n*80]                             *NOTE
 //SORTOUT  DD  UNIT=SYSDA,SPACE=(space2)
 //TMSRPT   DD  SYSOUT=A
 //SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
 //SYSOUT   DD  SYSOUT=A
 //SORTWK01 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK02 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK03 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK04 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK05 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK06 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //TMINPUT  DD  DSN=input,DISP=OLD
[//ACTIVITY DD  DSN=master,DISP=OLD  ]
[//DEFAULTS DD  DSN=defaults,DISP=OLD]
 //SYSWK01  DD  UNIT=SYSDA,SPACE=(space2)
 //SYSIN    DD  *
 (control statements here)
 /*
```

Note: The BLKSIZE can be any multiple of the LRECL, 80. If no BLKSIZE is specified, the BLKSIZE defaults to 800.

Parameter Definitions

The following parameters may be specified in any order in the PARM= field. Each parameter may be specified only once.

DATEFMT=(fmt)

Specifies the preferred *output* date format for date fields being converted. If no value is specified, DATEFMT=(DEFAULT) is used. This date format is also used as the SYSIN *input* date format, unless a different date format is specified in SYSIN. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

TMSEXPDT

Identifies the output record format. Required.

TMINPUT=[*1*|nn]

Relates to the TMINPUT data set. Cannot be used with DEFAULTS=nn and/or MASTER=nn parameters. The value of nn specifies the number of physical records that make one logical record.The default is 1.This parameter is optional.

UXCO=

Identifies the user exit load module name that TMSCONVR uses. This module can be renamed but the new name cannot be TMSUPDTE, TMSEXPDT or TMRECORD. The new name can be a maximum of eight bytes and must follow MVS naming conventions. This parameter is optional.

XITU=

Identifies the user exit load module name that TMSCONVR uses for alphanumeric volume serial number conversions. This module can be renamed but the new name cannot be TMSUPDTE, TMSEXPDT or TMRECORD. The new name can be a maximum of eight bytes and must follow MVS naming conventions. This parameter is optional. If CA 1 is active, the active version of XITU will be used instead of what is specified here.

MASTER=[*1*|nn]

Relates to the ACTIVITY data set. Cannot be used with the TMINPUT=nn parameter. The value of nn specifies the number of physical records that make one logical record. The default is 1. This parameter is optional.

DEFAULTS=[*1*|nn]

Relates to the DEFAULTS data set. Cannot be used with the TMINPUT=nn parameter. The value of nn specifies the number of physical records that make one logical record. The default is 1. This parameter is optional.

MAXERR=[*50*|nnnnn]

Specifies the number of errors accepted before the job takes a U0006 abend. The default is 50 and the maximum is 99999. This parameter is optional.

JCL Considerations

space1

Is the amount of space required for the number of records to be generated. The output record block size is 800 (80-byte records blocked by 10).

space2

Is the amount of space required for records 386-bytes long, blocked 10 records per block. There is approximately one output record for each logical input record.

TMINPUT or ACTIVITY DEFAULTS

The input data set may reside on any input device as a sequential data set. If it is a member of a PDS (partitioned data set), it must be called as DSN=dsname (member) in the JCL. If it is statement input, it must be called as *,DCB=BLKSIZE=80 in the JCL. If it is NL tapes, the DCB data must be provided.

An example PARM list is:

```
PARM='DATEFMT=(MMMDD YYYY),TMSEXPDT,TMINPUT=2,TMSUXCO'
```

TMSCONVR creates output records in TMSEXPDT format. All output dates are formatted according to the date pattern MMMDD YYYY. The TMINPUT data set is read twice to make one logical record. The TMSUXCO user exit is loaded and called for user data manipulation.

The TMSUXCO user exit can be used to generate partial data set names. The exit must place the character X in the one-byte field TMFLAG1 to identify those data sets which should be considered partial (driven by the leftmost characters) by the RDS.

Creating TMSUPDTE Output

Either the TMINPUT data set or the ACTIVITY and DEFAULTS data sets can be input to TMSCONVR to produce TMSUPDTE control statements. The VOL and REP control statements are generated. However, VER control statements are not generated. Omit unused DD statements. DD DUMMY is an active file.

We recommend that you take whatever precaution is necessary to ensure the accuracy of the volume serial number and the data being replaced.

The minimum data that is required as input to TMSCONVR is the data set name and volume serial number.

```
 //stepname EXEC PGM=TMSCONVR,
 //             PARM='parm'
 //STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
 //TMOUTPUT DD  DSN=tmsupdte.control.statements,DISP=(NEW,CATLG,DELETE),
 //             UNIT=SYSDA,SPACE=(space1),
 //             [DCB=BLKSIZE=n*80]                             *NOTE
 //SORTOUT  DD  UNIT=SYSDA,SPACE=(space2)
 //TMSRPT   DD  SYSOUT=A
 //SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
 //SYSOUT   DD  SYSOUT=A
 //SORTWK01 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK02 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK03 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK04 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK05 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //SORTWK06 DD  UNIT=SYSDA,SPACE=(TRK,(space),,CONTIG)
 //TMINPUT  DD  DSN=input,DISP=OLD
[//ACTIVITY DD  DSN=master,DISP=OLD  ]
[//DEFAULTS DD  DSN=defaults,DISP=OLD]
 //SYSWK01  DD  UNIT=SYSDA,SPACE=(space2)
 //SYSIN      DD    *
 (control statements here)
 /*
```

Note: The BLKSIZE can be any multiple of the LRECL, 80. If no BLKSIZE is specified, the BLKSIZE defaults to 800.

Parameter Definitions

You can specify the following parameters in any order in the PARM= field. You can specify each parameter only once.

DATEFMT=(fmt)

(Optional) Specifies the preferred *output* date format for date fields being converted. If no value is specified, DATEFMT=(DEFAULT) is used. This date format is also used as the SYSIN *input* date format, unless a different date format is specified in SYSIN. The date pattern that (fmt) indicates is limited to 10 bytes and must be enclosed in parentheses. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

TMSUPDTE

Identifies the output record format. This parameter is *required*.

TMSUXCO

(Optional) Identifies the user exit load module name that TMSCONVR should use. You can rename this module but the new name cannot be TMSUPDTE, TMSEXPDT, or TMRECORD. The name can be a maximum of 8 bytes and must follow MVS naming conventions.

TMINPUT=[*1*|nn]

(Optional) Relates to the TMINPUT data set. Cannot be used with DEFAULTS=nn and/or MASTER=nn parameters. The value of nn specifies the number of physical records that make one logical record.

Default: 1

MASTER=[*1*|nn]

(Optional) Relates to the ACTIVITY data set. Cannot be used with the TMINPUT=nn parameter. The value of nn specifies the number of physical records that make one logical record.

Default: 1

DEFAULTS=[*1*|nn]

(Optional) Relates to the DEFAULTS data set. Cannot be used with the TMINPUT=nn parameter. The value of nn specifies the number of physical records that make one logical record.

Default: 1

NOCHAIN

Causes NOCHAIN to be placed on the VOL control statement. This parameter indicates that the volume to be updated is the second or higher volume sequence of a multivolume data set.

NODSN

Causes NODSN to be placed on the VOL control statement. This parameter indicates that you do not have to verify the data set name.

SCRATCH

Causes NODSN and SCRATCH to be placed on the VOL control statement. This parameter indicates that updates to the TMC Volume record should be made only if the volume is in scratch status.

UNEXPIRE

Causes the UNEXPIRE keyword to be placed on TMSUPDTE volume control statements, allowing the scratch bit to be turned off when TMSUPDTE is later executed.

MAXERR=[*50*|nnnnn]

(Optional) Specifies the number of errors that are accepted before the job takes a U0006 abend. The default is 50 and the maximum is 99999.

Default: 50

Limits: 99999

HEXZEROS

(Optional) Causes the default values to be placed into fields that are not specified on the input files. See Keywords and Their Use (see page 199) for valid keywords. The assigning of default values is based upon the keywords that the input data sets do not use.

NOINTAL

Causes the NOINTAL keyword to be placed on TMSUPDTE volume control statements allowing the bypass of turning on the internally updated indicator.

JCL Considerations

space1

Is the amount of space that is required for the number of records to be generated. The output record block size is 800 (80-byte records blocked by 10).

space2

Is the amount of space that is required for records 386 bytes long, blocked 10 records per block. The product uses approximately one output record for each logical input record.

TMINPUT or ACTIVITY DEFAULTS

The input data set can reside on any input device as a sequential data set. If it is a member of a PDS (partitioned data set), it must be called as DSN=dsname (member) in the JCL. If it is statement input, it must be called as *,DCB=BLKSIZE=80 in the JCL. If it is NL tapes, the DCB data must be provided.

An example PARM list is:

```
PARM='TMSUPDTE,DATEFMT=(MMMDD YYYY),MASTER=02,NODSN,HEXZEROS'
```

TMSCONVR creates output records in TMSUPDTE format. All output dates are formatted according to the date pattern MMMDD YYYY. The ACTIVITY data set is read twice to make one logical record. The DEFAULTS data set is read one time to make one logical record. The VOL control statement in the output contains the keyword NODSN. Default values are assigned to fields not specified on the input files.

If there is no PARM list, TMSUPDTE default records are written to the TMOUTPUT data set. One physical record on the input data set is equal to one logical record.

Note: TMSCONVR does not create REP statements for the ZZUSER fields. If the statements are required, perform a manual update using TMSUPDTE.

Control Statement Specification

The control statement for TMSCONVR is SYSIN, which describes the TMINPUT or DEFAULTS data set and **must** be specified. The standards for this control statement are:

- All parameters for a keyword, except CLR and DATEFMT, must be specified.
- Each control statement must begin with a keyword in position 1.
- A keyword and its parameters must be contained on one statement.
- Maximum length of data on a control statement is 79 positions, followed by a delimiting blank.
- As many control statements as are needed may be used to describe the presence of acceptable input data.
- An asterisk (*) in position 1 indicates a comment statement.
- Any errors detected causes a U0006 abend.
- Keywords may not be specified more than once.

The table on Keywords and Their Use (see page 199) lists the valid keywords used on the SYSIN control statements and the ACTIVITY data set. The default values used if HEXZEROS are requested in the PARM list are also listed. Valid keywords for each type of output are indicated by an X beneath the format title. Abbreviations for keywords are enclosed in parentheses after the keyword.

SYSIN Control Statement Format

```
keyword=(loc,lng,format[,CLR][,DATEFMT=(fmt)]),...
```

Control Statement Definition

keyword

Is one of the valid keywords listed in Keywords and Their Use (see page 199). The keyword must be valid for the output record format requested.

loc

Is the displacement of the data in the input record relative to position 1.

lng

Is the length of the data in the input record.

format

Is the character representation of the data in the input record. The following are acceptable values for format:

BI

- binary

CH

- character

PD

- packed decimal

HEX

- hexadecimal

Note: Date fields input for conversion are only accepted in packed decimal or character format. See Input Restrictions for Date Fields (see page 184) for information regarding input restrictions.

CLR

Indicates to generate the default value when the field contains null data.

DATEFMT=(fmt)

Describes the input date format of a date field if it is different from the date format specified on the EXEC statement in the JCL. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

SYSIN Control Statement Examples

```
VOLSER=(2,6,CH),EXPDT=(8,4,PD),CJOB=(12,8,CH)
CDATE=(20,10,CH,CLR,DATEFMT=(YYYY/MM/DD))
```

Or

```
VOLSER=(2,6,CH)
EXPDT=(8,4,PD)
CJOB=(12,8,CH)
CDATE=(20,10,CH,CLR,DATEFMT=(YYYY/MM/DD))
```

The expiration date is input in packed format (yyddd) and requires no conversion. However, the creation date is input in character format and requires an appropriate date format because it is different from the preferred output date format specified earlier in the EXEC statement. The examples describe an input file with the following format:

Position	Meaning
2 through 7	Volume serial number
8 through 11	Expiration date
12 through 19	Creating job name
20 through 29	Creation date

Keywords and Their Use

```
Keywords            Default       TMSUPDTE  TMSEXPDT  TMRECORD
---------------------------------------------------------------------------

ACCT                HEXZEROS          X                   X   (NOTES)
BATCHID             HEXZEROS                              X
BLKCNT              HEXZEROS          X                   X
BLKSIZE             HEXZEROS          X                   X
BTHDATE             ZEROS             X                   X   (NOTES
CDATE(C)            HEXZEROS          X                   X   (NOTES)
CDDNAME             PRE-TMS           X                   X
CJOB(J)             PRE-TMS           X         X         X
CLNCNT              HEXZEROS          X                   X
COUNT               HEXZEROS          X                   X
CPGM                HEXZEROS          X                   X
CSTEP               PRE-TMS           X                   X
CTIME               HEXZEROS          X                   X   (NOTES)
CUNIT               HEXZEROS          X                   X
DATECLN             ZEROS             X                   X   (NOTES)
DEN                 X'C3'             X                   X
DSN(D)              HEXZEROS          X         X         X
DSN17               PRE-TMS           X                   X
EXPDT(E)            60001             X         X         X
FLAG1               X'44'             X                   X
FLAG2               HEXZEROS          X                   X
FLAG3               HEXZEROS          X                   X
LABEL               X'02'             X                   X
LDATE               ZEROS             X                   X   (NOTES)
LJOB                PRE-TMS           X                   X
LPGM                HEXZEROS          X                   X
LRECL               HEXZEROS          X                   X
```

Notes:

- This field for TMRECORD defaults to packed zeros.
- This field for TMRECORD defaults to blanks.

```
Keywords            Default       TMSUPDTE  TMSEXPDT  TMRECORD
---------------------------------------------------------------------------

LTIME               HEXZEROS          X                   X
LUNIT               HEXZEROS          X                   X
NEXTVOL             HEXZEROS          X                   X
NUMDSNB             HEXZEROS          X                   X
OUTCODE(0)          HEXZEROS          X                   X
OUTDATE             ZEROS             X                   X    (NOTE)
PRERRC              HEXZEROS          X                   X
PRERRI              HEXZEROS          X                   X
PREVVOL             HEXZEROS          X                   X
PWERRC              HEXZEROS          X                   X
PWERRI              HEXZEROS          X                   X
RECFM               HEXZEROS          X                   X
RETPD                                           X
ROBID               X'40'             X                   X
ROBTY               X'40'             X                   X
SLOT                HEXZEROS          X                   X
SMSMC               HEXZEROS          X         X         X
TRERRC              HEXZEROS          X                   X
TRERRI              HEXZEROS          X                   X
TRTCH               X'80'             X                   X
TWERRC              HEXZEROS          X                   X
TWERRI              HEXZEROS          X                   X
USECLN              HEXZEROS          X                   X
VOLSEQ(V)           1                 X                   X
VOLSER              HEXZEROS          X                   X
WRETPD                                          X
1STDSNB             HEXZEROS                              X
1STVOL              HEXZEROS          X                   X
```

Note: This field for TMRECORD defaults to packed zeros.

TMS Report 18 - User-Supplied Input Converted to TMRECORD Format

```
MONDAY, JULY 18, 2005.190                        Y O U R   C O M P A N Y   N A M E                                ROxxxxx PAGE=00001

TMS REPORT-18    USER-SUPPLIED INPUT CONVERTED TO TMRECORD FORMAT



    EXPDT=(51,4,PD)
    CRTDT=(117,4,PD)
    VOLSER=(1,6,CH)
    DSN=(7,44,CH)
    CJOB=(125,8,CH)
```

```
MONDAY, JULY 18, 2005.190                        Y O U R   C O M P A N Y   N A M E                                   PAGE=00002

TMS REPORT-18    USER-SUPPLIED INPUT CONVERTED TO TMRECORD FORMAT

             5 RECORDS READ BY SYSIN

            18 RECORDS READ BY TMINPUT  (LOGICAL RECORD COUNT)

            18 RECORDS WRITTEN TO TMOUTPUT
```

TMS Report 18 - User-Supplied Input Converted to TMSEXPDT Format

```
MONDAY, JULY 18, 2005.190                        Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT-18    USER-SUPPLIED INPUT CONVERTED TO TMSEXPDT FORMAT



    EXPDT=(18,4,PD)
    *RTDT=(27,10,CH,DATEFMT=(MMMDD YYYY))
    *OLSER=(1,6,CH)
    DSN=(8,09,CH)
    CJOB=(38,8,CH)
```

```
MONDAY, JULY 18, 2005.190                        Y O U R   C O M P A N Y   N A M E                                   PAGE=00002

TMS REPORT-18    USER-SUPPLIED INPUT CONVERTED TO TMSEXPDT FORMAT

             5 RECORDS READ BY SYSIN

             4 RECORDS READ BY TMINPUT  (LOGICAL RECORD COUNT)

             4 RECORDS WRITTEN TO TMOUTPUT
```

TMS Report 18 - User-Supplied Input Converted to TMSUPDTE Format

```
FRIDAY, AUGUST 28, 2005.243                      Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT-18    USER-SUPPLIED INPUT CONVERTED TO TMSUPDTE FORMAT



    DSN=(1,5,CH)
    CJOB=(20,8,CH)
    CDATE=(30,10,CH,DATEFMT=(DDD/YYYY))
```

```
FRIDAY, AUGUST 26, 2005.243                      Y O U R   C O M P A N Y   N A M E                                   PAGE=00002

TMS REPORT-18    USER-SUPPLIED INPUT CONVERTED TO TMSUPDTE FORMAT

             3 RECORDS READ BY SYSIN

             3 RECORDS READ BY ACTIVITY (LOGICAL RECORD COUNT)

             3 RECORDS READ BY DEFAULTS (LOGICAL RECORD COUNT)

            15 RECORDS WRITTEN TO TMOUTPUT
```

TMSCONVR Report Field Definitions

Control statement image

Input control statement

Message

Processing message

RECORDS READ BY SYSIN

Number of input control statements processed from SYSIN DD

RECORDS READ BY TMINPUT (LOGICAL RECORD COUNT)

Number of records processed for input from TMINPUT DD

RECORDS READ BY ACTIVITY (LOGICAL RECORD COUNT)

Number of records processed from ACTIVITY DD

RECORDS READ BY DEFAULTS (LOGICAL RECORD COUNT)

Number of records processed from DEFAULTS DD

RECORDS WRITTEN TO TMOUTPUT

Number of records processed for output to TMOUTPUT DD

TMSCOPY Utility—Back Up/Restore the TMC and Audit Data Set

Use the TMSCOPY utility to back up the TMC and Audit data set transactions written since the previous backup.

TMSCOPY may also be used to restore the TMC if it is corrupted or destroyed.

CA 1 must be active or batch active to run TMSCOPY to *back up* the TMC.

CA 1 must be batch active or inactive to *restore* the TMC.

Report Description

TMSCOPY produces TMS Report 31 - Backup/Restore the TMC File. A sample report showing the backup operation appears on TMS Report 31 - Backup the TMC File (see page 211), and a sample report showing the restore operation appears on TMS Report 31 - Restore the TMC File (see page 211).

Backing Up the TMC and Audit Data Set

The TMC and Audit data set should be backed up at least once daily by running TMSCOPY. The TMC is backed up first, then the Audit records written since the previous backup are written to tape. There is no need to stop tape processing during TMSCOPY backup processing.

The Audit data set is a direct-access, *wraparound* data set that must be backed up regularly, or eventually it will fill up. Every update to a TMC record, and certain other tape processing and CA 1 initialization activity, is recorded in the Audit data set. During backup processing, all Audit records written since the previous backup are copied to tape, and logical pointers in the Audit Control Record are reset. Audit records are never physically cleared of information; rather, as new records are written, the new records eventually wrap and overlay older records.

The integrity of the Audit is critical to the TMC restoration process. When the number of Audit records written exceeds a site-defined level, CA 1 issues warning messages to the system console indicating a TMC backup is required. At this point any online/update requests and CA 1 utilities, except TMSCOPY and TMSINIT, will abend with a U0008 error. Should the Audit reach full capacity, CA 1 will abend (S7yy) all jobs requesting tape I/O until TMSCOPY has successfully completed.

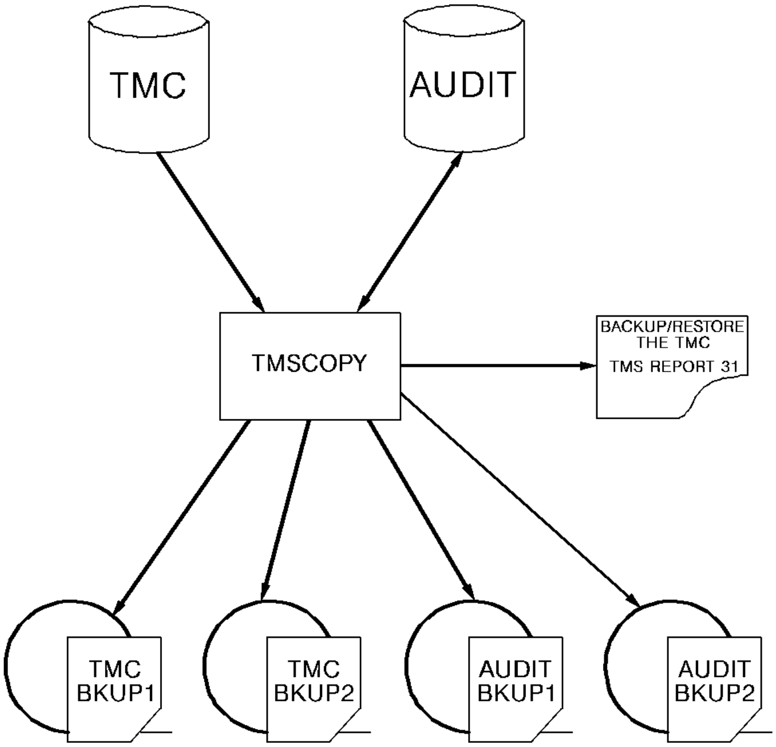
A parameter option in the TMOOPTxx member of *hlq*.CTAPOPTN, THAUDT, establishes the warning threshold limit on the number of Audit records written. This limit is a percentage of the total number of Audit records which can be written before the first IEFTMS9 TMC BACKUP REQUIRED warning message is issued. The system default for THAUDT is 80 percent. When the Audit threshold is reached during normal system processing, the highlighted IEFTMS9 message prompts the operator to backup the TMC. The message includes the percentage of the Audit file that has been utilized. As the utilization increases, a new highlighted IEFTMS9 message is issued with the updated percentage and the previous IEFTMS9 message is deleted. Once the TMC backup has completed, the Audit file utilization drops back to zero and the last IEFTMS9 message is deleted.

The Audit backup tapes contain only those Audit records written since the previous execution of TMSCOPY. This data should be retained for 6 to 12 months to aid in problem analysis, as a problem may go undetected for an extended period of time. TMS Report 31 - Backup/Restore the TMC File should also be saved for at least six months, because it is useful in trend analysis. The records that are written to tape from a BLOCKSET audit data set are not compatable with programs prior to the enhancement. The enhanced programs allow a restore using backups of either or both types of audit data set.

Assuming that TMSCOPY is run once daily, you may want to consider MODing the daily Audit backup files to a weekly Audit file, then MODing the weekly Audit data to a monthly file to minimize the number of tapes involved. For suggested procedures see the *Administration Guide*.

TMSCOPY creates two backups of the TMC as shown in the figure below; the second backup should be stored in an off-site vault. It is a good idea to run TMSCOPY prior to execution of the Vault Management System (VMS) so that an alternate copy of the TMC can be moved off-site with your other data sets (the data set name of this alternate backup must be defined to VMS).

The following diagram illustrates Backing Up TMC and Audit Data with TMSCOPY:



Job Control Statements

```
//stepname EXEC PGM=TMSCOPY,
//             PARM='[DATEFMT=(fmt),]BACKUP'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//DUMPDD   DD  DSN=CAI.CA1.TMCBKUP.TAPE1(+1),UNIT=TAPE,
//             [DCB=BLKSIZE=n*340,]FREE=CLOSE,               *NOTES
//             DISP=(NEW,CATLG,DELETE),LABEL=EXPDT=99000
//DUMPDD2  DD  DSN=CAI.CA1.TMCBKUP.TAPE2(+1),UNIT=TAPE,
//             DISP=(NEW,CATLG,DELETE),
//             [DCB=BLKSIZE=n*340,]FREE=CLOSE,               *NOTES
//             LABEL=EXPDT=99000
//AUDUMP   DD  DSN=CAI.CA1.AUDBKUP(+1),UNIT=TAPE,
//             DISP=(NEW,CATLG,DELETE),
//             [DCB=BLKSIZE=n*370,]FREE=CLOSE,               *NOTES
//             LABEL=EXPDT=99000
//AUDUMP2  DD  DSN=CAI.CA1.AUDBKUP2(+1),                     OPTIONAL
//             DISP=(NEW,CATLG,DELETE),
//             UNIT=TAPE,
//             [DCB=BLKSIZE=n*370,]FREE=CLOSE,                *NOTES
//             LABEL=EXPDT=99000
```

Notes:

- The BLKSIZE can be any multiple of the LRECL, which is 340. If no BLKSIZE is specified and the release of the operating system supports system determined blocksize, the operating system will select the blocksize, otherwise the BLKSIZE for DUMPDD and DUMPDD2 defaults to 32640 for tape or 8840 for DASD devices.
- The BLKSIZE can be any multiple of the LRECL, 370. If no BLKSIZE is specified and the release of the operating system supports system determined blocksize, the operating system will select the blocksize, otherwise the BLKSIZE for AUDUMP defaults to 32560 for tape or 8880 for DASD devices.
- The BLKSIZE can be any multiple of the LRECL, 370. If no BLKSIZE is specified and the release of the operating system supports system determined blocksize, the operating system will select the blocksize, otherwise the BLKSIZE defaults to 32560 for tape or 8880 for DASD. This DDNAME is OPTIONAL.

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 31. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

BACKUP

Indicates the TMC and Audit data sets are to be backed up. This is the default.

Restoring the TMC and Audit Data Set

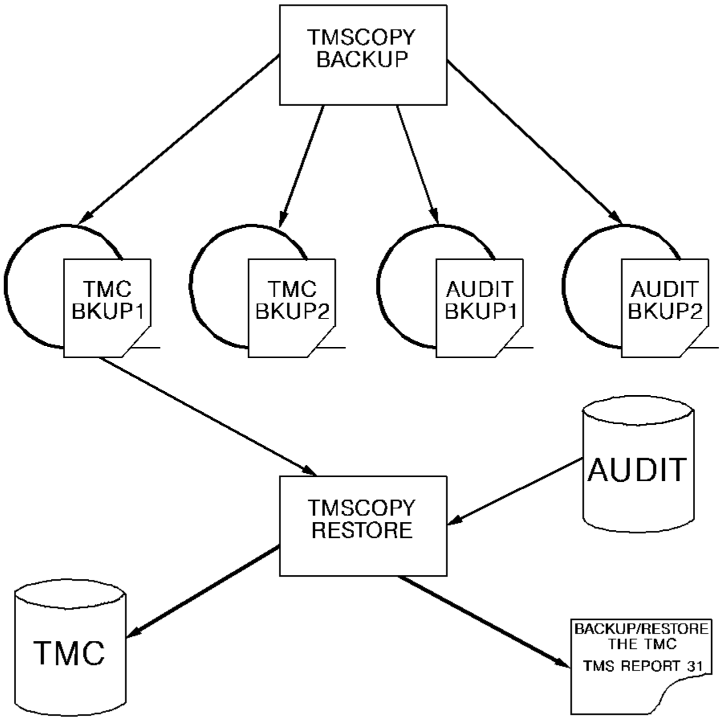
In the event of a complete loss of the Audit data set, it should be reformatted with TMSFORMT. CA 1 should then be reinitialized using TMSINIT on *all* CPUs. For complete instructions, see the procedure on Formatting the Audit Data Set (see page 262).

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 which were made since the last backup, as illustrated in the figure below. Only Audit record types 0, 1, and 2 are used in a restore.

Note: For information on TMC/Audit and Disaster Recovery, see the *Administration Guide*.

If no current or alternate backup tape is available, see Restoring the TMC with a Noncurrent Backup (see page 209).

The following diagram illustrates Restoring the TMC:



Important! A DASD RESTORE of the device holding the TMC effectively destroys the TMC itself unless followed immediately by a TMSCOPY RESTORE of the TMC. The TMC should be allocated as unmovable: DSORG=PSU. PSU should prevent the TMC and Audit from being moved during a DEFRAG operation. If disk volumes containing the TMC or Audit are reorganized or compressed, the TMC or Audit files **must** be excluded. The TMC and Audit files should also be excluded from Archive lists.

****DO NOT RUN OTHER TAPE JOBS DURING THE TMSCOPY RESTORE****

Various console messages may be issued during TMSCOPY BACKUP or RESTORE processing. These messages are documented in the *Messages Reference Guide*.

Restoring the TMC with the Current Backup

When TMSCOPY is used to restore the TMC, an additional DD statement called RESTART is required. Should the program abend, the data set referenced by RESTART is cataloged and referenced again in any subsequent attempts to execute the TMSCOPY restore operation.

If the TMC has been completely destroyed (all tape jobs abend or CA 1 initialization abends), execute all three steps defined in the job control statements after requesting that the CTS address space release the TMC.

Note: For more information, see the *Administration Guide*.

In STEP 1, reply to the IEFTMS0 WTOR with the CA 1 system password to deactivate CA 1. In STEP 3, activate CA 1 by replying U.

If the TMC contains invalid data (for example, a DASD restore has restored an old TMC), respond to the IEFTMS0 WTOR to batch activate CA 1. CA 1 must be batch active or inactive to perform the restore. If CA 1 is batch active during a restore, the restore tape is sequence checked and the volume serial number and data set name are compared to the current backup volume serial number and data set name stored in the Control Record on the Audit data set. A return code of 12 is issued if the backup tape is not the most current, primary backup.

```
//* STEP1 WILL DEACTIVATE OR BATCH ACTIVATE CA 1
//STEP1    EXEC PGM=TMSINIT
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TEMPAUDT DD  UNIT=SYSDA,
//             SPACE=(TRK,(10,1)),DCB=BLKSIZE=n*96              *NOTES
//TMSPARM  DD  DSN=hlq.CTAPOPTN,DISP=SHR
//* STEP 2 WILL RESTORE THE TMC
//STEP2    EXEC PGM=TMSCOPY,
//             PARM='[DATEFMT=(fmt),]RESTORE'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//DUMPDD   DD  DSN=latest.backup(either.copy).of.TMC,DISP=OLD
//TMC      DD  DSN=CAI.CA1.TMC,DISP=SHR
//AUDITDD  DD  DSN=CAI.CA1.AUDIT,DISP=SHR
//RESTART  DD  DSN=dsname,DISP=(MOD,DELETE,CATLG),
//             UNIT=SYSDA,SPACE=(TRK,1),
//             DCB=(RECFM=FB,BLKSIZE=n*370)                     *NOTES
//* STEP3 WILL REACTIVATE CA 1
//STEP3    EXEC PGM=TMSINIT
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TEMPAUDT DD  UNIT=SYSDA,
//             SPACE=(TRK,(10,1)),DCB=BLKSIZE=n*96              *NOTES
//TMSPARM  DD  DSN=hlq.CTAPOPTN,DISP=SHR
```

Notes:

- The BLKSIZE can be any multiple of the LRECL, 96.
- The BLKSIZE can be any multiple of the LRECL, 370.
- See CTAPSAMP member TMSCOPYR for a disaster recovery restore using a backup of the AUDIT file on the same tape.

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 31. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

RESTORE

Indicates the TMC is to be restored and the Audit data set is to be used to post all entries which were made since the last backup was taken.

RESTORE,NOAUDIT

Indicates the TMC is to be restored but the Audit data set is not to be used to update the TMC. Even though Audit records will not be applied, the AUDIT DD statement in the restore JCL must reference a validly formatted Audit data set. This option should not be used without a thorough understanding of the Audit file. You may wish to contact CA 1 Technical Support before using this PARM.

JCL Considerations

RESTART DD

The RESTART DD defines a data set that is to be cataloged if an abend occurs during RESTORE processing. This DD is required for RESTORE operations. If the abend is a system abend such as S813, simply perform the required adjustments, such as a data set name correction, and resubmit TMSCOPY.

A commonly encountered CA 1 abend in RESTORE processing is a U0010. This abend occurs when TMSCOPY encounters a Type 5 Audit record, which is created whenever the TMC has been structurally modified by TMSFORMT or TMSREMOV.

If the reason for the RESTORE is that a TMSFORMT extend operation or TMSREMOV purge operation was invalid, the U0010 abend can be considered normal. Delete the RESTART data set, and execute TMSCOPY to back up the restored TMC image.

In some instances, it may be necessary to capture Audit data written subsequent to the TMSFORMT or TMSREMOV operation. You are advised to contact CA 1 Technical Support in this instance.

Restoring the TMC with a Noncurrent Backup

In the event you have to restore the TMC with a backup other than the most current backup or the most current alternate backup, you are urged to contact the CA 1 Technical Support Group for advice before starting the RESTORE operation.

CA 1 must be *deactivated* prior to a noncurrent RESTORE and the CTS address space must RELEASE the TMC (see the *Administration Guide*). TMSCOPY restore JCL uses an additional DD statement to read in all Audit data from the backup tapes created since the noncurrent version of the TMC (the version that is used as input). A parameter of NONCURR must be specified after RESTORE.

The following JCL illustrates a noncurrent restore using a minus 3 (-3) generation of the TMC backup; the minus 2, 1 and current generation of the Audit backups; and the live Audit file. The TMC image is written first; then, in chronological order, the Audit data from tape is written, followed by the Audit data from the live file.

```
//* STEP1 WILL DEACTIVATE CA 1
//STEP1    EXEC PGM=TMSINIT
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TEMPAUDT DD  UNIT=SYSDA,
//             SPACE=(TRK,(10,1)),DCB=BLKSIZE=n*96              *NOTES
//TMSPARM  DD  DSN=hlq.CTAPOPTN,DISP=SHR
//* STEP 2 WILL RESTORE THE TMC
//STEP2    EXEC PGM=TMSCOPY,
//             PARM='[DATEFMT=(fmt),]RESTORE,NONCURR'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//DUMPDD   DD  DSN=minus.three.backup.of.TMC(-3),DISP=SHR
//TMC      DD  DSN=CAI.CA1.TMC,DISP=SHR
//AUDITBK  DD  DSN=minus.two.backup.of.AUDIT(-2),DISP=SHR       *NOTES
//         DD  DSN=minus.one.backup.of.AUDIT(-1),DISP=SHR
//         DD  DSN=current.backup.of.AUDIT(0),DISP=SHR
//AUDITDD  DD  DSN=CAI.CA1.AUDIT,DISP=SHR
//RESTART  DD  DSN=dsname,DISP=(MOD,DELETE,CATLG),
//             UNIT=SYSDA,SPACE=(TRK,1),
//             DCB=(RECFM=FB,BLKSIZE=n*370)                     *NOTES
//* STEP3 WILL REACTIVATE CA 1
//STEP3    EXEC PGM=TMSINIT
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TEMPAUDT DD  UNIT=SYSDA,
//             SPACE=(TRK,(10,1)),DCB=BLKSIZE=n*96              *NOTES
//TMSPARM  DD  DSN=hlq.CTAPOPTN,DISP=SHR
```

Notes:

- The BLKSIZE can be any multiple of the LRECL, 96.
- The AUDITBK DD properly references the Audit backup tapes, oldest to newest, in the concatenation. Failure to concatenate correctly or failure to read the tapes in the proper sequence (oldest tape first, most current tape last) may cause the TMC to contain incorrect information.
- The BLKSIZE can be any multiple of the LRECL, 370. Once complete, CA 1 can be reactivated using TMSINIT, and the DBS subtask can be started again or directed to OBTAIN the TMC again through CTS address space commands. to back up the restored image.

Control Statement Specification

There are no control statements for TMSCOPY.

TMS Report 31 - Backup the TMC File

```
THURSDAY, JUNE 30, 2005.179                      Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT-31     BACK-UP/RESTORE THE TMC FILE

                         TMC HAS BEEN SUCCESSFULLY BACKED UP.
                         CONTROL RECORD #4 HAS BEEN WRITTEN.

  STATISTICS FROM TMC:
                         NUMBER OF TMC RECORDS BACKUP........12,000
                         NUMBER OF DSNB RECORDS BACKUP.......25,000
                         NUMBER OF ACTIVE DSNB'S.............6,842
                         NUMBER OF AVAILABLE DSNB'S.........18,158

  STATISTICS FROM AUDIT:
                         NUMBER OF AUDIT RECORDS............25,000
                         NUMBER OF AUDIT RECORDS BACKUP......18,700
                         DATE OF LAST BACKUP.............MMMDD YYYY
                         TIME OF LAST BACKUP..................09:35
                         VOLUME SERIAL NUMBER................103871
                         1ST AUDIT RECORD ON TAPE...........0000001
                         NEXT AVAILABLE RECORD..............0018701
                         NUMBER OF BATCH UPDATE RECORDS.......7,270
                         NUMBER OF INPUT RECORDS.............4,800
                         NUMBER OF OUTPUT RECORDS.............5,900
                         NUMBER OF DSNB RECORDS.................230
                         NUMBER OF EXCEPTION RECORDS.............30
                         NUMBER OF NOT-SCRATCH RECORDS...........40
                         NUMBER OF TYPE 5 RECORDS...............430

  LOCATION ON AUDIT FILE:
                                      RELATIVE      TOTAL
                            DATE       NUMBER      NUMBER

                         MMMDD YYYY    000002      001087
                         MMMDD YYYY    001089      011029
                         MMMDD YYYY    012118      000256
                         MMMDD YYYY    012374      000569
                         MMMDD YYYY    012943      005759
```

TMS Report 31 - Restore the TMC File

```
THURSDAY, AUGUST 25, 2005.242                    Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT-31     BACK-UP/RESTORE THE TMC FILE




         BACK-UP TAPE SHOULD BE  - VOL=SER=103871  DSN=CAI.CA1.TMC.COPY
         BACK-UP TAPE BEING USED - VOL=SER=103871  DSN=CAI.CA1.TMC.COPY



                   47 TMC  RECORDS RESTORED
                  200 DSNB RECORDS RESTORED

                    3 AUDIT RECORDS APPLIED
```

TMSCOPY Report Field Definitions

NUMBER OF TMC RECORDS BACKUP

Number of Volume records backed up

NUMBER OF DSNB RECORDS BACKUP

Number of DSNB records backed up

NUMBER OF ACTIVE DSNB'S

Number of used DSNB records backed up

NUMBER OF AVAILABLE DSNB'S

Number of unused DSNB records backed up

NUMBER OF AUDIT RECORDS

Number of Audit records formatted in Audit

NUMBER OF AUDIT RECORDS BACKUP

Number of Audit records backed up

DATE OF LAST BACKUP

Date of TMC backup updated in Audit Control Record

TIME OF LAST BACKUP

Time of TMC backup updated in Audit Control Record

VOLUME SERIAL NUMBER

Volume serial number of TMC backup updated in Audit Control record

1ST AUDIT RECORD ON TAPE

First Audit record number backed up

NEXT AVAILABLE RECORD

Next Audit record to be written to Audit

NUMBER OF BATCH UPDATE RECORDS

Number of Type 0 Audit records backed up

NUMBER OF INPUT RECORDS

Number of Type 1 Audit records backed up

NUMBER OF OUTPUT RECORDS

Number of Type 2 Audit records backed up

NUMBER OF DSNB RECORDS

Number of DSNB Audit records backed up

NUMBER OF EXCEPTION RECORDS

Number of Type 3 Audit records backed up

NUMBER OF NOT-SCRATCH RECORDS

Number of Type 4 Audit records backed up

NUMBER OF TYPE 5 RECORDS

Number of Type 5 records backed up

DATE

Date of Audit records in Audit

RELATIVE RECORD

Relative record number of first Audit record for this date in Audit

TOTAL NUMBER

Number of Audit records for this date in Audit

BACK-UP TAPE SHOULD BE -

TMC backup VOLSER and data set name that should be used in the restore operation

BACK-UP TAPE BEING USED -

TMC backup VOLSER and data set name used in the restore operation

TMC RECORDS RESTORED

Number of Volume records restored from TMC backup volume

DSNB RECORDS RESTORED

Number of DSNB records restored from TMC backup volume

AUDIT RECORDS APPLIED

Number of Audit records applied from Audit

TMSCTLG Utility—Remove Tapes from Catalog Control

Note: The functionality of this utility has been incorporated into TMSCLEAN. However, if all systems sharing the TMC have not been upgraded to r5.2 or higher and the TMSOSCAT utility with a PARM=SYNC has not been run on all systems sharing the TMC; this utility should still be used.

Once all systems sharing the TMC have been upgraded to r5.2 or higher and the TMSOSCAT utility with a PARM=SYNC has been run on all systems sharing the TMC; the OCTLG control statement (in the TMOOPTxx member of CTAPOPTNS) can be set to N and this utility no longer needs to be executed.

TMSCTLG resets the expiration date in the TMC for all volumes under Catalog Control (CATALOG or CATLG in control statements and online interfaces, or LABEL=EXPDT=99000, or ACCODE=xCACATLG in JCL) that no longer have corresponding entries in the system catalog. The date is set to the current system date plus any catalog extension (defined by option R9 in the TMOOPTxx member of *hlq*.CTAPOPTN). If the tape has multiple files and a secondary file has an expiration date value higher than this calculated date, the higher expiration date value is used.

For multifile tapes, the volume set will remain protected as long as any file in the volume set still exists on the system catalog. Secondary files are expired by Catalog Control once it is determined that a corresponding entry does not exist on the system catalog. When no entries remain on the system catalog for all files in the volume set, the expiration date is reset in the Volume record for File 1.

To protect long-running tape jobs, a two-day minimum hold limit exists for tapes under Catalog Control. Because CA 1 sets the expiration date at tape OPEN and the system does not post the entry to the system catalog until step termination, it is possible that, during the execution of TMSCTLG, a tape that is still on the drive could *appear* to be eligible for expiration.

To avoid the premature scratching of tapes under these circumstances, TMSCTLG checks first to see if the volume has been cataloged. (The X'80' bit is turned on in FLAG2.) The r5.2 or higher realtime catalog interface will turn on the X'80' bit in Flag 2 as soon as the tape is cataloged in the OS Catalog. If not, TMSCTLG keeps the tape for two days from its creation date (CDATE). DSNBs are kept for two days from their CDATE even if uncataloged.

Catalog Days Control (CATLG/ddd in control statements and online interfaces, or LABEL=EXPDT=90ddd in JCL) can be used to control the amount of time to wait before TMSCTLG will process the data set. Catalog Days Control specifies that the data set is retained for ddd days. Once ddd days have passed (calculated as CDATE plus the ddd value), TMSCLEAN will automatically change the expiration date of the data set to CATALOG, allowing subsequent runs of TMSCTLG to process the data set.

TMSCTLG should be executed before TMSCLEAN to determine the scratch eligibility of tapes under Catalog Control.

CA 1 must be active or batch active to run this utility.

TMSCTLG operates in one of the following modes:

OS Catalog Mode

To run TMSCTLG in the OS catalog mode, all system catalog(s) must be available to the system on which TMSCTLG is run. A LOCATE is issued to determine the status of the tape data set in the system catalog. This applies to File 1 data sets and secondary data sets. The information returned by LOCATE determines whether the tape data set is still under Catalog Control or needs to be expired by Catalog Control.

If all catalogs are not available to the system on which TMSCTLG is executed, then this mode of operation should not be used. Data sets which have CATALOG as their expiration date may be prematurely expired.

If TMSCTLG is run in the OS catalog mode, then it is the only required step for Catalog Control processing. OS catalog mode is the default in the provided JCL member JOBCTLG, and is controlled by the absence of the JCL DDNAME VWORK.

IDCAMS Listing Mode

This mode of operation for TMSCTLG is available for multi-CPU installations in which system catalog(s) 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 data set name, file sequence, volume sequence, and media type, is extracted, sorted, and written to a VSAM data set in a separate procedure prior to executing TMSCTLG. Once this VSAM file (defined by the JCL DDNAME VWORK) is created, it is used 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.

To help ensure all IDCAMS listings are provided to the listing extract job CTSLSTEX as subsequent input to the TMSCTLG step as the VWORK VSAM file, Catalog Listing Verification can be performed. Catalog Listing Verification takes a list of expected system catalog data set names, defined in the SYSIN DDNAME JCL statement, and verifies that the VSAM VWORK data set contains entries from those system catalogs. If any system catalog data set coded in SYSIN is not found or is available but does not contain any tape data sets, TMSCTLG will terminate with a condition code of 12. This eliminates the accidental exclusion of a system catalog IDCAMS listing to CTSLSTEX. If the SYSIN DDNAME JCL is not present, Catalog Verification is bypassed. If you do not want to do Catalog Verification, do not code //SYSIN DD DUMMY. This type of coding will result in a condition code 12 and a verification error.

If Listing mode is chosen for Catalog Control processing, it requires three steps. The first step is to run either IDCAMS on each system which contains a system catalog that may have tape data sets cataloged. After listings have been obtained from all system catalogs needed, the listing data sets are used as input to the second step.

The second step is a procedure which first executes CTSLSTEX to extract data set information from the IDCAMS listing files then sorts and creates the VSAM data set. The provided JCL for this step is in *hlq*.CTAPJCL member TMSLSTEX.

The third step in the Listing mode procedure is execution of TMSCTLG, using the VSAM data set created in the second step. This VSAM data is defined by the JCL DDNAME VWORK. Optionally, SYSIN can be used to provide the list of expected catalog names for Catalog Listing Verification.

Special Considerations

Take care to choose the correct mode for an installation. If OS catalog mode is chosen but not all system catalogs were available to the system on which TMSCTLG ran, or the Listing mode is chosen and not all the catalogs were verified, TMSCTLG will expire the data sets not found using either mode. To correct this situation, TMSCLEAN can be executed with a PARM=RESETCAT to return any nonscratched data sets to Catalog Control. For TMC Volume records, the expiration date is reset to CATALOG, and the TMECATLG bit (X'08') in FLAG2 is turned off. For TMC DSNB records (secondary files), the expiration date is reset to CATALOG, and the DSNBECAT bit (X'20') in DSNB FLAG1 is turned off. *This method of recovery must be run before TMSCLEAN expires the data sets.*

TMSCTLG expires secondary files (DSNB records) once it is determined that they are no longer cataloged, and that the create date is at least 2 days ago. The Catalog Control expiration date (current date plus the R9 option extension), is moved to DSNBEXDT and the DSNBECAT indicator (X'20') is turned on in DSNB FLAG1. This allows you to view the DSNB records belonging to the volume set and see which files are still cataloged. In either mode, TMSCTLG treats data sets cataloged with a file sequence of zero the same as file sequence one.

TMSCTLG ends with a return code equal to the highest return code from the LOCATE macro indicating that a LOCATE error occurred.This applies to the OS catalog mode only.

A flat file in the CA 1 Retention Record format is created for use with a report writer in a subsequent step. For TMSCTLG OS catalog mode, the return code from the LOCATE macro is provided in the RDUNCIND field of the Retention Record in the format LE=xx.

User Exit

A user exit is available for tailoring TMSCTLG processing. User exit TMSXCTLG is invoked prior to each TMC Volume record and DSNB record update. The user exit is called only if the option XCTLG is set to the name of the exit itself in the TMOOPTxx member of *hlq*.CTAPOPTN. The Retention Record for the TMC or DSNB being processed is presented to the user exit. TMSXCTLG can CONTINUE processing, BYPASS the update, or MODIFY the expiration date and FLAG1 or DSNB FLAG1 setting.

Note: For more information on this user exit see the *Programming Guide*.

Job Control Statements

```
//stepname EXEC PGM=TMSCTLG,
//             [,PARM='[TEST][,DEBUG][,HOLDDAYS=nn]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//*SYSIN   DD  *                                     *NOTES
//*VWORK   DD  DSN=cai.ksds.data.set,DISP=SHR        *NOTES
//SYSPRINT DD  SYSOUT=A
//EARLOUT  DD  DSN=earlout.catalog.file,             *NOTES
//             DISP=(NEW,PASS,DELETE),
//             UNIT=SYSDA,
//             VOL=SER=xxxxxx,
//             SPACE=(TRK,(10,5)),
//             DCB=(RECFM=FB,LRECL=376,BLKSIZE=15416)
```

Notes:

- SYSIN is used only for Listing mode.
- VWORK is used only for Listing mode.
- EARLOUT is the flat file created using the CA 1 Retention Record format used to create the TMS Report 09 in a subsequent step using a report writer such as CA Earl. See CA Earl program TMECTLG in the CAI.CTAPECPB library.

Parameter Definitions

TEST

Process normally, but do not update any TMC records.

DEBUG

Used for debugging purposes only.

HOLDDAYS=nn

Allowable values are 2-99. Used to alter the standard 2 day hold that must elapse after a tape has been created before TMSCTLG is allowed to expire it. To protect long running tape jobs, the HOLDDAYS retention is applied to all tape data sets under catalog control, which were never cataloged.

Listing Mode Only - Step 1

For TMSCTLG Listing mode, IDCAMS must be run on all systems containing catalogs for tape data sets.

Use the following JCL to list your ICF catalogs.

```
//stepname EXEC PGM=IDCAMS
//SYSPRINT DD  DSN=idcams.listing,
//             UNIT=SYSDA,
//             DISP=(NEW,PASS,DELETE),
//             SPACE=(CYL,5),
//             DCB=(LRECL=125,RECFM=VB,BLKSIZE=629)
//SYSIN    DD  *
 LISTCAT       NONVSAM ALL
```

If multiple IDCAMs are to be processed, the SYSPRINT should use DISP=(MOD,CATLG).

Listing Mode Only - Step 2

Once the listing data sets are created for each system containing catalogs for tape data sets, the following JCL found in member TMSLSTEX of *hlq*.CTAPJCL creates the VSAM Listing mode data set:

```
//TMSLSTEX JOB (ACCOUNT)
//*-------------------------------------------------------------------*
//*                                                                   *
//*        THIS JOB IS USED TO PROCESS IEHLIST/IDCAMS LISTINGS        *
//*        AS INPUTS TO PRODUCE AN OUTPUT DATA SET (VSAM KSDS)        *
//*        WHICH IS TO BE USED AS INPUT TO TMSCTLG (VWORK DD).        *
//*                                                                   *
//*        EDIT IN THE IDCAMS OUTPUT LISTING DATA SETS AND            *
//*        THE VSM KSDS DATA SET NAME BELOW IN THE JCL.               *
//*                                                                   *
//*-------------------------------------------------------------------*
//CTSLSTEX PROC LINKLIB='CAI.CTAPLINK',
//             OUTPUT='cai.ksds.data.set',
//             SOUT='*',
//             TGTSER=,
//             TGTUNIT='SYSDA',
//             WRKUNIT='SYSDA'
//STEP1    EXEC PGM=IDCAMS
//SYSPRINT DD  SYSOUT=&SOUT.
//DD1      DD  VOL=SER=&TGTSER.,
//             UNIT=&TGTUNIT.,
//             DISP=SHR
//*
```

```
//STEP2    EXEC PGM=CTSLSTEX
//STEPLIB  DD  DSN=&LINKLIB.,
//             DISP=SHR
//LISTOUT  DD  DSN=&.&TEMP.,
//             DISP=(NEW,PASS),
//             UNIT=&WRKUNIT.,
//             SPACE=(CYL,5)
//IDCAMS   DD  DSN=idcams.listing,
//             DISP=SHR
//SYSOUT   DD  SYSOUT=&SOUT.
//SYSPRINT DD  SYSOUT=&SOUT.
//*
//STEP3    EXEC PGM=SORT
//SYSOUT   DD  SYSOUT=&SOUT.
//SORTIN   DD  DSN=&.&TEMP.,
//             DISP=SHR
//SORTOUT  DD  DSN=&OUTPUT.,
//             DISP=SHR
//SORTWK01 DD  UNIT=SYSDA,
//             SPACE=(TRK,(1,1))
//SORTWK02 DD  UNIT=SYSDA,
//             SPACE=(TRK,(1,1))
//SORTWK03 DD  UNIT=SYSDA,
//             SPACE=(TRK,(1,1))
//         PEND
//CTSLSTEX EXEC CTSLSTEX
//STEP1.SYSIN  DD *
 DELETE (cai.ksds.data.set) CLUSTER -
        PURGE FILE (DD1)
 DEFINE CLUSTER                                    -
        (NAME('cai.ksds.data.set') VOLUMES(volser) -
               KEYS(58 0) RECSZ(120 120))  -
        INDEX                                      -
               (NAME('cai.ksds.data.set.INDEX') CYL(5 5)) -
        DATA                                       -
               (NAME('cai.ksds.data.set.DATA') CYL(5 5) -
                KEYS(58 0) CISZ(X'1000'))
/*
//STEP3.SYSIN  DD *
 SORT FIELDS=(1,58,CH,A,61,4,PD,D)
 SUM FIELDS=NONE
/*
//
```

Control Statement Specification

For the Listing mode of TMSCTLG, expected system catalog data set names can be provided for Catalog Verification. These catalog data set names are provided in the JCL DDNAME SYSIN. SYSIN has a record length of 80. The standards for this control statement are:

- one system catalog data set per statement.
- data set name can begin in any column.
- comments are indicated by an asterisk (*) in column 1.

SYSIN Control Statement Format

The SYSIN is a list of the catalogs to be verified by TMSCTLG. The control statement should not start in column 1.

2

```
your.catalog.name
```

SYSIN Control Statement Format Example:

```
//SYSIN DD *
    ICF.VMVXE01
    ICF.VMVXE02
```

TMS Report 83 - TMSCTLG Messages

TMS Report 83 lists processing messages by TMSCTLG.

```
SUNDAY, FEBRUARY 27, 2005.056                 Y O U R   C O M P A N Y   N A M E                              PAGE=00001

TMS REPORT-83          TMSCTLG Messages

Parms Received: TEST

Options in effect:   TMCDSN CAI.CA1.TMC
                     AUDDSN CAI.CA1.AUDIT
                     XCTLG  Y
                     R9     0000002

Processing mode is: OS Catalog Lookups

Opening DDNAME EARLOUT: SYS95058.T102517.RA000.JOBCTLG.REPORTS
                        TSO76A

TMC  records using Catalog Control       =3
DSNB records using Catalog Control       =0
TMC  records using Catalog/DDD           =0
DSNB records using Catalog/DDD           =0

TMC  records expired by Catalog Control  =3
DSNB records expired by Catalog Control  =0

Number of updates bypassed by TMSXCTLG   =0
Number of modifications by TMSXCTLG      =0
Number of data set lookups issued        =2

Total Records Not Updated (TEST)         =3
Total Records to Output File             =3
```

TMS Report 09 - Expired Catalog Listing

The following example shows a partial listing of volumes that have been removed from Catalog Control.

The totals produced in the previous example reflect the number of volumes removed from Catalog Control, the total number of data sets under Catalog Control in the TMC, and information regarding the total number of LOCATEs issued and LOCATE errors. If any bad multivolume pointers are indicated, execute TMSPTRS to analyze the chains in the TMC and correct the reported errors. TMECTLG can be used to produce the following report.

```
SUNDAY, FEBRUARY 27, 2005.056                      Y O U R   C O M P A N Y   N A M E                        PAGE      1

                                                 PARM=TEST

TMS REPORT-09        EXPIRED CATALOG LISTING                                   DATA SETS WILL EXPIRE YYYY/058

------------------------------------------------------------------------------------------------------------------------------
VOLSER     VOL   DATA SET NAME                              CREATE     CREATE         CREATE   LAST       LAST USE   UNCATALOG
           SEQ                                              JOBNAME    DATE             TIME   JOB        DATE       INDICATOR
------------------------------------------------------------------------------------------------------------------------------
980006      1    CAI.CA1.MULTIVOL                           JOBTAPER   YYYY/034     13:09:00   JOBTAPER   YYYY/034   LE=08
980007      2    CAI.CA1.MULTIVOL                           JOBTAPER   YYYY/034     13:09:00   JOBTAPER   YYYY/034   LE=08
980016      1    CAI.CA1.TAPER.OU01SL01                     JOBTAPER   YYYY/054     23:07:00   JOBTAPER   YYYY/054   LE=08
END OF REPORT
```

TMSCTLG Report Field Definitions

Parms Received: xxxxxxxx

Parameters passed to program

Options in effect:

CA 1 options and values used by program

Processing mode is: xxxxxxxx

Processing mode used by program

Opening DDNAME EARLOUT: xxxxxxxx

Data set name for EARLOUT DD

TMC records using Catalog Control

Number of Volume records using Catalog Control

Number of volume records with bad CDATE

Number of Volume and DSNB records with a bad creation date

DSNB records using Catalog Control

Number of DSNB records using Catalog Control

TMC records using Catalog/DDD

Number of Volume records using Catalog Days Control

DSNB records using Catalog/DDD

Number of DSNB records using Catalog Days Control

TMC records expired by Catalog Control

Number of Volume records expired by Catalog Control

DSNB records expired by Catalog Control

Number of DSNB records expired by Catalog Control

Number of dataset lookups issued

Number of data set LOCATEs issued

Number of data set locate errors

Number of data set LOCATE errors encountered

Total Records to Output File

Number of Volume and DSNB records written to EARLOUT DD

Total Records Not Updated (TEST)

Number of Volume and DSNB records that would have been updated if not TEST MODE

Number of updates bypassed by TMSXCTLG

Number of TMC updates bypassed by TMSXCTLG user exit

Number of updates modified by TMSXCTLG

Number of TMC records modified by TMSXCTLG user exit prior to TMC update

PARM=xxxxxxxx

Parameters passed to program

DATA SETS WILL EXPIRE xxxxxxxxxx

Expiration date to which tape volumes are reset

VOLSER

Volume serial number of tape volume

VOL SEQ

Volume sequence number of tape volume

DATA SET NAME

Data set name of tape volume

CREATE JOBNAME

Creating job name of tape volume

CREATE DATE

Creation date of tape volume

CREATE TIME

Creation time of tape volume

LAST JOB

Last job name using tape volume

LAST USE DATE

Last used date of tape volume

UNCATALOG INDICATOR

Processing message

TMSCYCLE Utility—Remove Tapes from Cycle Control

Use the TMSCYCLE utility to select 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 File 1 EXPDT only.

To remove volumes from Cycle Control, TMSCYCLE updates the expiration date in the TMC to the current date plus the number of days defined in the TMOOPTxx member of *hlq*.CTAPOPTN (RC parameter). TMSCYCLE does not consider the generation/version number for GDG data sets. Therefore, GDGs under Cycle Control may expire prior to meeting the GDG limit. If Cycle Control is used for ABARS pseudo GDG data sets (ending in ".CnnVnnnn"), system option ADGDG should be set to YES. This allows the pseudo GDGs to be treated the same way as regular GDG data sets. 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. To determine individual creations, CDATE, CTIME, and CJOB are considered. 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 will have separate cycles. When set to DSN, the creating job name is not considered by TMSCYCLE.

CA 1 considers all like tape data sets created during the same day to be one cycle if the system parameter option CDAY=DAY was specified in the TMOOPTxx member of *hlq*.CTAPOPTN. If the system parameter options CDAY=DAY and CJOB=JOB were specified in the TMOOPTxx member of hlq.CTAPOPTN, then different jobs creating the same data sets may be cycled differently and each tape data set may be considered a separate cycle. If you suspect that too many cycles are being held, verify the system options specified.

TMSCYCLE should be executed before TMSCLEAN to determine the scratch eligibility of tapes under Cycle Control.

CA 1 must be active or batch active to run this utility.

User Exit

A user exit is available for tailoring TMSCYCLE processing. TMSXCYCL is invoked prior to the TMC update. The user exit is called only if the option UXCYCL is set to the name of the exit itself in the TMOOPTxx member of *hlq*.CTAPOPTN. The Retention Record is presented to the user exit. TMSXCYCL can continue processing, bypass the update, or modify the expiration date. For more information see the *Programming Guide*.

Cycle Abuse

Cycle abuse is the difference in EXPDT values between two Cycle Controlled tapes with the same DSN. For example, the Cycle Control for a data set is CYCLE/003 (three cycles). The keyword was erroneously changed for one creation of a data set to CYCLE/030 (30 cycles). The one data set created with CYCLE/030 keeps 30 tapes under Cycle Control until the 31st cycle is created. With a cycle abuse limit of 27 or less, TMSCYCLE reports this data set as abusing the cycle limit.

Report Description

TMS Report 16 - Possible Cycle Abuse Listing and TMS Report 16 - Expired Cycle Control Listing report on the volumes expired by TMSCYCLE. Samples of these reports appear on TMS Report 16 - Expired Cycle Control Listing (see page 227).

The Possible Cycle Abuse Listing identifies those data sets for which CA 1 has detected cycle abuse and is produced first. The Expired Cycle Control Listing reports on the volumes removed from Cycle Control.

Job Control Statements

```
//stepname EXEC PGM=TMSCYCLE
//             [,PARM='[,ABUSELMT=nnn][,TEST]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//SYSOUT   DD  SYSOUT=A
//SORTOUT  DD  UNIT=SYSDA,SPACE=(CYL,2)
//TMSRPT   DD  SYSOUT=A
//EARLOUT  DD  DSN=&.&CHANGES.,DISP=(,PASS),    *NOTE
//             UNIT=SYSDA,SPACE=(CYL,2),
//             DCB=(RECFM=FB,LRECL=376,BLKSIZE=15416)
//SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK02 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK06 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
```

Note: The EARLOUT DD should be used as input to TMECYCL or other program to print the Expired Cycle Control Listing (Figure 1-46).

The BLKSIZE of the EARLOUT data set can be any multiple of the LRECL, which is 376.

Parameter Definitions

ABUSELMT=nnn

Cycle abuse limit (see the previous cycle abuse description). The value of nnn can be from one to three numeric digits in the range 1 to 363. The default cycle abuse limit of 1 is recommended. This parameter is optional.

TEST

Process normally, but do not update any TMC records. The Expired Cycle List is produced, but the TMC is not updated. This parameter is optional.

Control Statement Specification

There are no control statements for TMSCYCLE.

TMS Report 84 - TMSCYCLE Messages

The TMSCYCLE Messages report shows the parameters specified, options in effect for TMSCYCLE, and processing totals. If any discrepancies for data sets using different Cycle Control Values (cycle abuse) are found, they are listed on Report 84.

```
SUNDAY, FEBRUARY 27, 2005.056                    Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT-84           TMSCYCLE MESSAGES

 PARAMETER LIST: TEST,ABUSELMT=1


OPTIONS IN EFFECT: TMCDSN  CAI.CA1.TMC
                   AUDDSN  CAI.CA1.AUDIT
                   UXCYCL  Y
                   CDAY    TAPE
                   CJOB    DSN
                   RC      0000000

OPENING DDNAME EARLOUT: SYS94058.T101516.RA000.JOBCYCLE.REPORTS
                        ON VOLUME MVXE90

                   61 RECORDS READ
                    0 RECORDS BYPASSED BY USER EXIT
                    0 RECORDS MODIFIED BY USER EXIT
                    4 RECORDS WOULD HAVE BEEN UPDATED
```

TMS Report 16 - Expired Cycle Control Listing

The Expired Cycle Control Listing shows which tapes have been removed from cycle control. This listing is printed by CA Earl program TMECYCLE.

```
SUNDAY, FEBRUARY 27, 2005.056                      Y O U R   C O M P A N Y   N A M E                        PAGE      1

                                                 PARM=TEST,ABUSELMT=1

TMS REPORT-16       EXPIRED CYCLE CONTROL LISTING                           VOLUMES WILL EXPIRE YYYY/058

------------------------------------------------------------------------------------------------------------------------------
VOLSER     VOL  DATA SET NAME                               CYCLE       CREATE     CREATE         CREATE   LAST       LAST USE
           SEQ                                              EXPDT       JOBNAME    DATE             TIME   JOB        DATE
-------------------------------------------------------------------------------------------------------------------------------
980008      1   CAI.CA1.DATA                                CYCLE/002   CA1TEST    YYYY/033     12:03:00   CA1TEST    YYYY/033
980005      1   CAI.CA1.DATA                                CYCLE/002   JOBTAPER   YYYY/034     11:58:00   JOBTAPER   YYYY/034
980006      1   CAI.CA1.DATA                                CYCLE/002   JOBTAPER   YYYY/044     13:09:00   JOBTAPER   YYYY/044
980007      2   CAI.CA1.DATA                                CYCLE/002   JOBTAPER   YYYY/044     13:09:00   JOBTAPER   YYYY/044

END OF REPORT
```

TMSCYCLE Report Field Definitions

PARAMETER LIST: xxxxxxxx

Parameters passed to program

Options in effect: xxxxxxxx

CA 1 options and values used by this program

Opening DDNAME EARLOUT: xxxxxxxx On volume xxxxxx

Data set name and volume for the EARLOUT DD

RECORDS READ

Number of volume records read from TMC

RECORDS BYPASSED BY USER EXIT

Number of volume record updates bypassed by TMSXCYCL user exit

RECORDS MODIFIED BY USER EXIT

Number of volume records modified by TMSXCYCL user exit prior to TMC update

UPDATED

Number of volume records updated

RECORDS WOULD HAVE BEEN UPDATED

Number of volume records that would have been updated if not TEST mode

PARM=xxxxxxxx

Parameters passed to program

VOLUME WILL EXPIRE xxxxxxxxxx

Expiration date to which tape volumes are reset

VOLSER

Volume serial number of tape volume

VOL SEQ

Volume sequence number of tape volume

DATA SET NAME

Data set name of tape volume

CYCLE EXPDT

Expiration date of tape volume

CREATE JOBNAME

Creating job name of tape volume

CREATE DATE

Creation date of tape volume

CREATE TIME

Creation time of tape volume

LAST JOB

Last job name using tape volume

LAST USE DATE

Last used date of tape volume

TMSDATA Utility—Produce TMS Sequential Data Sets

Use the TMSDATA program to produce sequential data sets from TMC Volume records and data set name block (DSNB) records.

You may use these data sets as input for the Generalized Report Writer (TMSGRW), or for programs written in-house. If a sort sequence is not specified by control statements, the TMC Volume records are in volume serial number sequence and the DSNB records are in DSNB number sequence. Inactive volume records and unused DSNB records are not processed.

CA 1 may be active, batch active or inactive when TMSDATA is run.

Report Description

TMSDATA produces TMS Report 36 - TMS Sequential Data Sets which list the sort sequence of TMC Volume records and DSNB records. A sample of this report appears on page TMS Report 36 - TMS Sequential Data Sets (see page 232).

Job Control Statements

```
//stepname EXEC PGM=TMSDATA
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//SYSUT1   DD  DSN=CAI.CA1.TMC,DISP=SHR
//SYSUT3   DD  DSN=tapetmc,UNIT=SYSDA,DISP=(NEW,CATLG,DELETE),
//            SPACE=(CYL,5)[,DCB=BLKSIZE=n*340]                    *NOTE
//SYSUT4   DD  DSN=tapedsnb,UNIT=SYSDA,DISP=(NEW,CATLG,DELETE),
//            SPACE=(CYL,5)[,DCB=BLKSIZE=n*170]                    *NOTE
```

Note: See the following JCL Considerations for the default block size.

If sort is requested, add the following:

```
//TMSRPT   DD  SYSOUT=A
//SORTLIB  DD  DSN=SYS1,SORTLIB,DISP=SHR
//SORTWK01 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),CONTIG)
//SORTWK02 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),CONTIG)
//SORTWK06 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),CONTIG)
//SYSIN    DD  *
(control statements here)
/*
```

JCL Considerations

SYSUT1 DD

Defines your TMC.

SYSUT3 DD

Defines the output data set that will contain the TMC volume records. The default block size is 32640 for tape or 8840 for DASD devices. The layout of this file can be found in the *Programming Guide*.

SYSUT4 DD

Defines the output data set that will contain the DSNB records. The default block size is 32640 for tape or 9010 for DASD devices. The layout of this file can be found in the Systems Programmer Guide under TMC DSN Record Definition.

Control Statement Definition

SORTTMC

Is the control statement identifier. It must begin in position 1 and be terminated by a space. SORTTMC instructs the program to sort the TMC Volume records in the specified sequence.

SORTDSNB

Is the control statement identifier. It must begin in position 1 and be terminated by a space. SORTDSNB instructs the program to sort the DSNB records in the specified sequence.

keyword

Valid keywords are listed in the *Programming Guide*. At least one keyword must be specified. A maximum of 20 keywords may be used with SORTTMC, and a maximum of 12 keywords may be used with SORTDSNB.

sequence

Defines the sequence order for the keyword. If specified, the sequence must be enclosed in parentheses. Valid sequences are:

A

Ascending order (default)

D

Descending order

If more than one keyword (sequence) is required, they must be separated by a comma and there must be at least one blank following the last keyword. If more than one statement is needed for one sort, repeat the SORT verb in the next statement and continue with the next keyword in the sort sequence.

Comment Statement

You may include comment statements by placing an asterisk (*) in position 1. Text for a comment may appear anywhere between positions 2 and 80 inclusive.

If SYSIN is defined but no control statements are in the JCL, the program executes with default values assigned, and the following message accompanies the report:

```
SORT WAS *NOT* REQUESTED
```

Example

```
SORTTMC DSN,J,CDATE
SORTDSNB DSN,J,CDATE
```

In this example, the TMC Volume records are spooled to the data set specified by SYSUT3 and the DSNB records are spooled to the data set specified by SYSUT4. These control statements sort the TMC volume records in data set name (ascending), job name (ascending), and creation date (ascending) order. The DSNB records are in sequence by ascending data set name, job name and creation date.

TMS Report 36 - TMS Sequential Data Sets

This report reflects the control statements read by TMSDATA and the sort sequence used for the volume and DSNB record filed created.

```
SATURDAY, JUNE 26, 2005.174                Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-36 - TMS SEQUENTIAL DATA SETS

SORTTMC DSN(A)
SORTDSNB DSN(A)

THE TMC WILL BE SORTED IN THE FOLLOWING SEQUENCE
 SORT FIELDS=(007,44,CH,A),SIZE=E30000

THE DSNBS WILL BE SORTED IN THE FOLLOWING SEQUENCE
 SORT FIELDS=(031,44,CH,A),SIZE=E30000
```

TMSDATA Report Field Definitions

Control statement image

Input control statement.

TMSDELET Utility—Activate, Deactivate TMC Volume Records

Use the TMSDELET 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 range(s) of records that are currently in INACTIVE status
- Deactivate a Volume record or range(s) of records that are currently in scratch status

If a physical tape volume does not exist, the TMC entry for that volume can be placed in inactive status. When a volume is marked inactive, the TMC volume entry still exists but that volume does not appear on the Scratch and Clean List, Volume Serial List, and so forth. A record in inactive status will have the X'02' bit on in FLAG1.

When new tapes arrive, the tape Volume records can be activated and marked available for use in the TMC. To activate a volume or volume range to the TMC using TMSDELET, the volume entries must already exist in the TMC. TMSFORMT is used for the initial entry of the Volume record into the TMC. It is possible to activate new tapes using the TMSTPNIT initialization utility, which is discussed on TMSTPNIT (see page 410).

To determine what volume serial numbers are available to be assigned to the new cartridges or reels, generate a list of inactive volume serial numbers by using the LIST control statement. As these numbers are assigned, the TMC volume entries must be activated to make the cartridges or reels immediately available for use. The TMSCOPY backup utility should be executed before activating or deactivating a large number of records.

Individual records can also be activated or deactivated using the CA 1 online facilities (CA 1 ISPF or TIQ Online Inquiry/Update). A discussion of the online facilities is provided in the *Administration Guide*.

CA 1 must be active or batch active to run TMSDELET.

Report Description

TMSDELET produces TMS Report 13 - Delete/Add/List TMC Records. Samples of these reports appear on TMS Report 13 - Delete TMC Records (see page 235) and TMS Report 13 - List Available TMC Records (see page 236).

Job Control Statements

```
//stepname EXEC PGM=TMSDELET[,PARM=TEST]
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//SYSIN    DD  *
(control statements here)
/*
```

Parameter Definitions

TEST

No updates are made to the TMC, but output is still produced.

Control Statement Specification

There are three control statements for TMSDELET:

- DELETE indicates a scratch tape volume is to be marked as inactive (deleted).
- ADD indicates an inactive (deleted) volume is to be activated for use.
- LIST produces a list of available, unused volume serial numbers in the TMC that are in inactive status.

Multiple control statements may be entered for each program execution.

DELETE Control Statement

The DELETE control statement indicates a physical tape volume does not exist although there is a TMC entry for that volume serial number. DELETE followed by the volume serial number specifies that volume as nonexistent in the tape library. DELETE followed by two volume serial numbers specifies a range of volumes as nonexistent in the tape library. A volume *must* be in scratch status before it can be deactivated. The format for the DELETE control statement is:

```
DELETE vvvvvv[-vvvvvv]
```

Control Statement Definition

DELETE

Is the control statement identifier delimited by at least 1 space and may start anywhere within the first 20 positions.

vvvvvv

Is the volume serial number to be deactivated. If two volume serial numbers separated by a dash (-) or any other delimiter such as a comma are specified, all TMC records in scratch status with VOLSERs in this inclusive range are marked inactive. Tapes not in scratch status in this range are listed as errors on the output. Tapes already in an inactive (DELETE) status are not processed.

ADD Control Statement

The ADD control statement indicates an available volume serial number is assigned to a physical tape volume. ADD followed by one volume serial number marks that volume as physically existing in the tape library. ADD followed by two volume serial numbers marks a range of volumes as physically existing in the tape library.

All volumes for TMC records updated with the ADD control statement are immediately available as scratch tapes. The tape is made active and the scratch bit is turned on. The format for the ADD control statement is:

```
ADD vvvvvv[-vvvvvv]
```

Control Statement Definition

ADD

Is the control statement identifier, delimited by a space and may start in any position.

vvvvvv

Is the volume serial number to be activated. If two volume serial numbers separated by a dash (or any other delimiter, such as a comma) are specified, all TMC records with VOLSERs in this inclusive range are activated. Tapes not in inactive status in this range are listed as errors on the output.

LIST Control Statement

The LIST control statement is used to produce a list of available volume serial numbers in the TMC that are marked inactive.

The format for the LIST control statement is:

```
LIST
```

Control Statement Definition

LIST

Is the control statement identifier and may start in any position.

Control Statement Example

The following example will deactivate volumes 000200 through 000210, activate volumes A00150 through A00199, and list all inactive tape volumes.

```
DELETE 000200-000210
ADD A00150-A00199
LIST
```

TMS Report 13 - Delete TMC Records

In this example, the TMSDELET operation marked volume serial numbers in the range 600000-600999, which were already in scratch status, as inactive.

```
SATURDAY, JUNE 25, 2005.174                      Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT-13                               DELETE TMC RECORDS

         VOLSER


         600000
         600001
         600002
         600003
         600004
         600005
         600006
         600007
         600008
         600009


                 10 RECORDS MARKED AS DELETED
```

TMS Report 13 - Add TMC Records

In this example, the TMSDELET ADD operation marked volume serial numbers in the range 600000-600049 active in the TMC. These volumes are now available as scratch tapes.

```
THURSDAY, JUNE 25, 2005.179                        Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT-13                                 ADD TMC RECORDS

         VOLSER


         600000
         600001
         600002
         600003
         600004
         600005
         600006
         600007
         600008


                 9 RECORDS ADDED
```

TMS Report 13 - List Available TMC Records

The TMSDELET LIST operation identifies volume serial numbers 600050-600139 as available VOLSERs that are marked inactive in the TMC.

```
THURSDAY, JUNE 25, 2005.179                        Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT-13                                 LIST AVAILABLE TMC RECORDS

         VOLSER


         600050 600051 600052 600053 600054 600055 600056 600057 600058 600059 600060 600061 600062 600063 600064
         600065 600066 600067 600068 600069 600070 600071 600072 600073 600074 600075 600076 600077 600078 600079
         600080 600081 600082 600083 600084 600085 600086 600087 600088 600089 600090 600091 600092 600093 600094
         600095 600096 600097 600098 600099 600100 600101 600102 600103 600104 600105 600106 600107 600108 600109
         600110 600111 600112 600113 600114 600115 600116 600117 600118 600119 600120 600121 600122 600123 600124
         600125 600126 600127 600128 600129 600130 600131 600132 600133 600134 600135 600136 600137 600138 600139


                90 AVAILABLE TMC RECORDS
```

TMSDELET Report Field Definitions

VOLSER

List of Volume records processed

RECORDS MARKED AS DELETED

Number of Volume records inactivated (deleted)

RECORDS MARKED AS DELETED - TEST MODE ONLY

Number of Volume records that would have been inactivated (deleted)

RECORDS ADDED

Number of Volume records activated if not TEST mode

RECORDS ADDED - TEST MODE ONLY

Number of Volume records that would have been activated if not TEST mode

AVAILABLE TMC RECORDS

Number of Volume records in inactive (DELETE) status

ERRORS

Number of Volume records not processed due to errors

RECORDS BYPASSED

Number of Volume records not processed for inactive (DELETE) status because they are already in inactive (DELETE) status

TMSDISP Utility—Perform Batch Auxiliary Disposition Processing

Use the TMSDISP Batch Auxiliary Disposition feature to performs several functions which streamline special tape handling requests and eliminate most of the associated paperwork.

This feature can do the following:

- Communicate postprocessing requests directly to the console operator or tape librarian.
- Generate external tape label requests for cartridges or reels.
- Automatically mark tapes out-of-area.
- Uncatalog data set name entries from the system catalog.

All label requests are processed by the Online Label Interface.

Note: For information on the Online Label Interface, see the *Administration Guide*.

CA 1 must be active or batch active to run TMSDISP.

Using Batch Auxiliary Disposition

The CA 1 batch utility TMSDISP is used to perform Batch Auxiliary Disposition processing. Auxiliary Disposition can also be performed in *realtime* mode. **Note:** See the *Administration Guide* for a discussion of Realtime Auxiliary Disposition processing.

Batch Auxiliary Disposition processing requires the data set being processed to 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 may 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 will complete with a return code of 16 if any errors are encountered.

A DD statement *must* be used to reference either in-stream control statements or a data set containing the control statements. The default ddname supplied by CA 1 is TMSDISP; however, this can be changed if your installation standards require the use of a different ddname. For information on on changing the CA 1 system options see the *Programming Guide*.

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

```
IECTMS8 dddd.vvvvvv.ssss (area).message.text
```

Where:

- IECTMS8 is the message ID.
- dddd is the address of the tape VOLSER.
- vvvvvv is the volume serial number.
- ssss is the volume sequence number.
- (area) is the optional out-of-area code and, when used, is enclosed in parentheses for verification purposes.
- message.text is the text copied from the control statement.

Report Description

TMSDISP does not generate a report.

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* and precedes the optional Message control statement. The format for the DSN statement is:

```
1   5

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

Control Statement Definition

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 parameter is *required*. If the UNCATLG option is specified, the explicit name must be coded to uncatalog.

UNCATLG

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

LABEL

Indicates an external label is to be printed. This parameter is optional. If a label is requested, it is processed by the Online Label Interface.

Message Control Statement

The Message control statement defines the format and content of messages, and the route code. This control statement is optional, and multiple statements are allowed. The format for the Message control statement is:

```
123456789012345678901

ddname  xr OUT=xxxx,message.text
  or
ddname  xr message.text
```

Control Statement Definition

ddname

The ddname may be a maximum of eight characters and must start in column 1.

x

The placement of any nonblank character in position 9 will require the operator to delete the WTO from the console.

r

Identifies the route code(s) to which the WTO is issued (the default is 13 and must be in position 10). Possible values 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)

OUT=xxxx

The value of 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.

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 will automatically check 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.

Example

The following TMSDISP program could be executed as a 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
//
```

Note: If the number of volumes for a data set exceeds 20, and ICF catalogs are used to retrieve the VOLSER, only the first 20 volumes are processed due to a limitation of LOCATE functions in TMSDISP.

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 may be any value; it does not have to be the actual ddname used in the PAYMASTR JCL.

TMSEXPDT Utility—Reset Expiration Dates in the TMC

Use the TMSEXPDT utility 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 ACCODE parameter, 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 or SMS Management Class.

Data sets that received the default retention defined by the RP parameter can be identified by:

- TMC Volume records - FLAG3 = TMDFEXU (bit X'04')
- TMC DSNB records   - DSNB FLAG1 = DSNBDFXU (bit X'04')

The RDS is a sequential file containing 80-byte records describing the SMS management class, the data set name, optional qualifying job name, the desired EXPDT or RETPD to be assigned, or an EXPDT/RETPD to be assigned to tape data sets closed by ABEND processing. Data set name and job name can be specified using the CA 1 Pattern Masking Specification. RDS entries are sorted according to the CA 1 pattern masking hierarchy, as established by CTSPM.

Note: For more information on pattern masking see the *Programming Guide*.

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 during OPEN processing when the JCL specifies an EXPDT, RETPD, or ACCODE parameter, thereby preventing RDS override action.

TMSEXPDT provides an option for reassigning an EXPDT or RETPD to a tape data set closed by ABEND processing. This allows tape data sets created in a step that abended to receive an EXPDT or RETPD that differs from the EXPDT or RETPD assigned to a tape data set that went through normal CLOSE processing.

In addition, TMSEXPDT can be used to reset the EXPDT or RETPD of all tape data sets matching an RDS rule. When SELECT=ALL is specified on the RDS rule, all matching data sets can be assigned a new EXPDT or RETPD, regardless of the eligible for RDS override bit in FLAG1.

CA 1 also provides an alternate way to enforce data set retention standards defined in the RDS. The Realtime Expiration feature is designed to apply the RDS rules in realtime as tape files are created. Note that the Retention Override (RO) parameter discussed above must also be set on for Realtime Expiration to be able to override expiration dates set through JCL. For further information on the Realtime Expiration feature see the information on the TMSINIT utility (see page 328) later in this guide.

Because TMSEXPDT is used to establish expiration dates in the TMC, TMSEXPDT must be executed before TMSCTLG, TMSCYCLE and TMSCLEAN.

CA 1 must be active or batch active to run TMSEXPDT.

For normal processing, TMSEXPDT queries the TMC to determine if an RDS override should be applied. Records that are eligible for normal TMSEXPDT processing can be identified by:

TMC Volume record

FLAG1 equal to TMDFAULT (bit X'01') (eligible for RDS override)
FLAG1 not equal to TMDELET (bit X'02') (volume in inactive (DELETE) status)

FLAG1 not equal TMSCRTCH (bit X'04') (volume in scratch status)
FLAG1 not equal TMABEND (bit X'10') (volume closed by abend)
FLAG1 not equal TMINTAL (bit X'80') (internal field manually updated)
FLAG2 not equal TMETMS (bit X'01') (expired by TMS, CA 11, or EDM)

DSNB records

DSNB FLAG1 equal to DSNBDFLT (bit X'01') (eligible for RDS override)
DSNB FLAG1 not equal DSNBUSRU (bit X'80') (updated by user)

DSNB FLAG1 not equal DSNBTMSI (bit X'40') (expired by TMS, CA 11, or EDM)
DSNB FLAG1 not equal DSNBABND (bit X'10') (data set closed by abend)

For abend date processing, records that are eligible for TMSEXPDT processing can be identified by:

TMC Volume records

FLAG1 equal to TMABEND (bit X'10')
FLAG1 equal to TMDFAULT (bit X'01')

FLAG1 not equal to TMDELET (bit X'02')
FLAG1 not equal TMSCRTCH (bit X'04')

FLAG1 not equal TMINTAL (bit X'80')
FLAG2 not equal TMETMS (bit X'01')

DSNB records

DSNB FLAG1 equal to DSNBABND (bit X'10')
DSNB FLAG1 equal to DSNBDFLT (bit X'01')

DSNB FLAG1 not equal DSNBUSRU (bit X'80')
DSNB FLAG1 not equal DSNBTMSI (bit X'40')

For reassigning EXPDT or RETPD to all data sets (SELECT=ALL), records eligible for TMSEXPDT processing can be identified by:

TMC Volume records

FLAG1 not equal TMDELET (bit X'02')
FLAG1 not equal TMSCRTCH (bit X'04')

DSNB records

any value in DSNB FLAG1

Special Considerations

A flat file in the CA 1 Retention Record format is created for use with a report writer in a subsequent step such as CA Earl. The RDRDSMSG field contains TMSEXPDT processing information such as the file sequence number of the data set with the highest expiration date and lets you know whether a matching RDS rule was found.

User Exit

User exit TMSXEXP is provided for tailoring TMSEXPDT processing. This exit is invoked prior to each TMC Volume record and DSNB record update. The user exit is called only if the option UXEXP is set to the name of the exit itself in the TMOOPTxx member of *hlq*.CTAPOPTN. The Retention Record being processed is presented to the user exit. TMSXEXP can continue processing, bypass the update, or modify the expiration and FLAG1 or DSNB FLAG1 setting.

Note: For more information see the *Programming Guide*.

Report Description

TMSEXPDT produces TMS Report 82 - TMSEXPDT MESSAGES. The EARLOUT DDNAME produces a flat file in the CA 1 Retention Record format which can be used to produce TMS Report 22 - CA 1 Default Expiration Date Listing and TMS Report 22 - CA 1 Batch Expiration Update Listing.

The Default Expiration Update Listing reports on all data sets that are eligible for RDS override and their current RDS expiration dates or retention periods. If the Retention Data Set (RDS) is a null data set (SYSIN DD DUMMY), the EARLOUT data set will contain data sets eligible for RDS override processing.

The Batch Expiration Update Listing created from the EARLOUT data sets shows any overrides performed.

CA Earl member TMEXPDT, which can be found in the CA 1 Source library (CAI.CTAPECPB), can be used to print the reports from the EARLOUT data sets.

Job Control Statements

```
//stepname EXEC PGM=TMSEXPDT
//             [,PARM='[DATEFMT=(fmt)][,DSNBS][,TEST][,NODFLT]
//             [,DEBUG][,TESTRDS][,LISTRDS]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//SYSPRINT DD  SYSOUT=A
//EARLOUT  DD  DSN=save.tmsexpdt.earlout,
//             DISP=(NEW,PASS,DELETE),
//             UNIT=SYSDA,
//             VOL=SER=VOLSER,
//             SPACE=(CYL,5),
//             DCB=(RECFM=FB,LRECL=376,BLKSIZE=15416)
//SYSIN    DD  *
 (RDS control statements here)
/*
```

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 82. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

Note: If a date pattern is defined and then subsequently revised, it may be necessary to reflect the pattern change in the optional preferred date value for EXPDT. If a DATEFMT other than Julian is specified, then the DATEFMT used in the RDS data set must match the DATEFMT used on the PARM statement. Read the caution text on DSN Control Statement Format and then revise the preferred date pattern, as necessary.

DSNBS

Indicates that DSNBs (secondary data sets) that are eligible for RDS override are to be processed. This option causes the volume expiration date to be reset to the new DSNB expiration date if the new RDS date is numerically higher than the volume expiration date. This parameter is optional.

TEST

Process normally, but do not update any TMC records.The EARLOUT data set contains records used for the Batch Expiration Update Listing, but the TMC is not updated. This parameter is optional.

NODFLT

Indicates that the EARLOUT data set will list records with an RDS match only.This parameter is optional.

DEBUG

This parameter can be used to examine the internal entries created from the Retention Data Set (RDS).This parameter is optional.

LISTRDS

This parameter will cause all the RDS rules to be printed, not just with those errors.

TESTRDS

Check the syntax of RDS rules only. The TMC is not examined. No records are written to the EARLOUT ddname. May only be specified with DEBUG or DATEFMT parameter.

JCL Considerations

SYSIN

Defines the Retention Data Set (RDS). It must be composed of fixed-length, 80-byte records. It may be included in the job stream in statement form, stored as a sequential data set, or exist as a member of a partitioned data set. SYSIN DD DUMMY will generate an EARLOUT data set of all data sets which have the eligible for RDS override bit on in FLAG1.

Control Statement Specification

The DSN or MGMTCLAS control statement is used in the Retention Data Set (RDS) to identify data sets to be controlled and the desired retention. The DSN or the MGMTCLAS keyword indicates the beginning of an RDS rule. Control statement keywords may begin in any column,but cannot exceed column 71. Columns 72 through 80 are ignored. Control statements longer than 71 characters must be split across two lines, with the first line ending in a comma. A keyword and its value must be coded on the same line.

Note: You may specify both LABEL and ABEND with the same control statement. You **must** specify at least one. See examples that follow.

DSN Control Statement Format

```
{                 }{             {keyword       }}
{                 }{             {preferred date}}[,J=create.jobname      ]
{D=dsname[-]      }{,LABEL=EXPDT={Julian date   }}[,JOB=create.jobname    ]
{DSN=dsname[-]    }{,LABEL=RETPD=nnnn            }[,JOBNAME=create.jobname]
 MGMTCLAS=smsclass {,LABEL=WRETPD=nnnn           }[,SELECT=ALL           ]
 M=smsclass        {,ABEND=RETPD=nnnn            }
                   {,ABEND=WRETPD=nnnn           }
                   {,ABEND=EXPDT={keyword       }}
                                 {preferred date}
                                 {Julian date   }
```

Control Statement Definition

D= / DSN=

Specifies the data set name to be overridden. This data set name appears in the TMC record. For generation data sets, do not specify the GDG or relative generation number. If you are specifying the entire data set name for a GDG do not put a "-" at the end to mask the "GnnnnVnnn" portion on the name. The DSN keyword is required if the MGMTCLAS keyword is not specified. The DSN keyword may begin in any position, and may be specified using CA 1 pattern masking rules. If the data set name contains special characters (other than masking characters), it must be enclosed within single quotes, such as DSN='ABC$'. If any of the special characters in the data set name match a masking character, the character must be immediately preceded by a back slash (\). Reference the *Programming Guide*, CA 1 Pattern Masking Feature, for complete details. DSN may not be specified with MGMTCLAS.

M= / MGMTCLAS=

Specifies the SMS Management Class found in the TMC record to be used to assign retention. SMS Management Class is obtained during data set OPEN processing, updated using the batch utility TMSSMSUP, or updated manually. The MGMTCLAS keyword is required if the DSN keyword is not specified. The MGMTCLAS keyword may begin in any position, and is eight characters maximum. MGMTCLAS may not be specified with DSN or JOB and must be fully qualified. Pattern masking may not be used.

LABEL=EXPDT= / ABEND=EXPDT=

Specifies an expiration date or hold criteria. This value can be a CA 1 keyword, a preferred date, or a Julian date. The expiration date specified must be greater than the system date at execution of TMSEXPDT. Delimiters are required if the date contains blanks or commas. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21). Expiration dates in the range of 98000 through 99365 are treated as Julian dates and are now invalid since the date is in the past. For more information on CA 1 keyword dates see Specifying CA 1 Keywords.

Important! When you specify a preferred date for EXPDT, any subsequent revision to the DATEFMT option in *hlq*.CTAPOPTN or to the DATEFMT=(fmt) parameter in the TMSEXPDT program JCL may result in an abend. Therefore, use *extreme* caution when revising either of these patterns.

You are strongly advised to change the date pattern of the preferred date for EXPDT so that it reflects the newly revised date pattern specified in hlq.CTAPOPTN or in the TMSEXPDT program JCL. This restriction applies only to the preferred date; if you are using a CA 1 keyword or a date in Julian format for EXPDT, this warning can be ignored.

LABEL=RETPD= / ABEND=RETPD=

Specifies the number of days to be added to the creation date to arrive at the new expiration date for the data set. If the creation date is zero, the system date is used.

LABEL=WRETPD= / ABEND=WRETPD=

Specifies the number of working days (which excludes weekends) to be added to the creation date to arrive at the new expiration date. Weekends are adjusted for in calculating the new expiration date. If the creation date is zero, the system date is used. WRETPD is a TMSEXPDT keyword and will cause a JCL error if used outside of the Retention Data Set. TMSEXPDT places the value specified on the control statement for WRETPD on TMS Report 22, not the calculated number of days retention.

J= / JOB= / JOBNAME=

Is the creating job name for the data set. This is an optional keyword and is used to qualify the RDS rule. Pattern masking specification may be used. Different expiration dates can be assigned to the same DSN created by different jobs. This keyword is not valid with MGMTCLAS, only DSN.

SELECT=ALL

Specifies that the RDS rule for the DSN or MGMTCLAS is to apply to all matching entries in the TMC. Selects and reassigns the expiration date based on the EXPDT or RETPD provided. This option can significantly increase processing time. SELECT is mutually exclusive with JOBNAME.

Comment Statement

You may include comment statements by placing an asterisk (*) in position 1. Text for a comment may appear anywhere between positions 2 and 71 inclusive. As of CA 1 r5.2 Service Pack 4, comments may also appear anywhere in the statement when enclosed in '/*....*/'. Both old and new style comments may appear after the RDS control parameters. At least one blank must separate the parameter from the comments.

```
Example 1: DSN=W-,LABEL=EXPDT='CATALOG' OLD STYLE COMMENTS
Example 2: DSN=W-,LABEL=EXPDT='CATALOG' /*NEW STYLE COMMENTS*/
Example 3: * ENTIRE STATEMENT WITH OLD STYLE COMMENTS
Example 4: /*ENTIRE STATEMENT WITH NEW STYLE COMMENTS*/
```

Control Statement Examples

```
DSN=CTI40.DFHOCANC,LABEL=EXPDT=DDMMMYYYY,ABEND=EXPDT=LDATE/001
DSN=CIF.DOS.V5M1MT,LABEL=EXPDT=CATLG
DSN='$TESTTMC',LABEL=WRETPD=30
DSN=CFCS.V42.RSERV,LABEL=EXPDT=PERM
DSN=CTI00.MAY88,LABEL=RETPD=30
DSN=CTI00.MAY88,LABEL=RETPD=45,JOB=CHILES
MGMTCLAS=INTERIM,LABEL=EXPDT=CATLG
M=BACKUPT,LABEL=EXPDT=CATLG
D=BACKUP-,LABEL=EXPDT=CATLG,J=WEEKLY
```

Note: The most explicit entries are processed first.

The example using LABEL=EXPDT=15APR2005 sets a preferred date expiration of April 15, 2005. LABEL=EXPDT=CATLG indicates the data set is to be placed under Catalog Control.

LABEL=WRETPD=30 adds 30 working days to the creation date of the data set to arrive at the new expiration date. Depending on the day of the week the tape was created, the data set in this example would be held between 40 and 42 days.

The example, LABEL=EXPDT=PERM, uses the CA 1 keyword PERM to specify permanent hold. The next two examples indicate that the user has separate hold criteria for the same data set name. This is accomplished by using the JOB parameter to qualify the data set name by its creating job name.

The last three examples indicate that data sets beginning with the partial data set name of BACKUP are to be assigned the expiration date coded. The JOB parameter is used to further qualify which data sets within that partial data set name group are to be assigned the permanent hold expiration date (PERM) or Catalog Control (CATLG).

TMS Report 82 - Retention Data Set (RDS)

```
SUNDAY, FEBRUARY 27, 2005.056                  Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-82               **   RETENTION DATA SET   **

PARMS RECEIVED: TEST



--> RETENTION INPUT RECORDS READ     24
--> RETENTION DEFINITIONS            6
--> RETENTION ERRORS FOUND           0
--> RETENTION DEFINITIONS IN ERROR   0
```

TMS Report 82 - TMSEXPDT Messages

```
SUNDAY,  FEBRUARY 27, 2005.056                   Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT-82          TMSEXPDT MESSAGES

OPTIONS IN EFFECT   TMCDSN  CAI.CA1.TMC
                    AUDDSN  CAI.CA1.AUDIT
                    UXEXP   Y
                    RP      0000005
                    RO      Y

OPENING DDNAME EARLOUT: SYS94058.T105050.RA000.JOBEXPDT.REPORTS
                        ON VOLUME TSO76A

** WARNING VOLSER=980008 DSN=CAI.CA1.TEST                                SYSTEM DATE USED FOR CREATION DATE

   0000006 RECORDS WRITTEN TO EARLOUT
   0000006 Default volumes selected
   0000000 DSNs eligible, no RDS entry
   0000000 Records modified by user exit
   0000000 Records bypassed by user exit
```

TMS Report 22 - CA 1 Default Expiration Date Listing

```
SUNDAY, FEBRUARY 27, 2005.056                      Y O U R   C O M P A N Y   N A M E                        PAGE      1

                                       PARM=TEST

TMS REPORT-22       CA 1 DEFAULT EXPIRATION DATE LISTING

-----------------------------------------------------------------------------------------------------------------------------
  VOLSER    VOL   FILE DATA SET NAME                            EXPIRATION CREATE   CREATE   CREATE     RDS        RDS MESSAGE
            SEQ    SEQ                                          DATE       JOBNAME  STEP     DATE       EXPDT
-----------------------------------------------------------------------------------------------------------------------------
  980005     1      1  CAI.CA1.DATA                             YYYY/072   JOBTAPER WRITESL  YYYY/034   YYYY/056   NO RDS REC
  980013     1      1  CAI.CA1.MVMF.DATA                        CYCLE/001  JOBTAPER WRITESL  YYYY/054   YYYY/056   NO RDS REC
  980014     2      1  CAI.CA1.MVMF.DATA                        CYCLE/001  JOBTAPER WRITESL  YYYY/054   YYYY/056   NO RDS REC
  980015     3      1  CAI.CA1.MVMF.DATA                        CYCLE/001  JOBTAPER WRITESL  YYYY/054   YYYY/056   NO RDS REC
  980016     1      1  CAI.CA1.TAPER.OU01SL01                   CYCLE/003  JOBTAPER WRITESL  YYYY/054   YYYY/061   NO RDS REC
  980003     1      1  CAI.CA1.TAPER.TEST                       LDATE/009  JOBTAPER WRITESL  YYYY/055   LDATE/003

END OF REPORT
```

TMS Report 22 - CA 1 Batch Expiration Update Listing

This report, created from the EARLOUT data set, lists data sets which were overridden by the RDS rules.

```
SUNDAY,     FEBRUARY   27, 2005.056                      Y O U R   C O M P A N Y   N A M E                      PAGE      1

                                       PARM=TEST

TMS REPORT-22       CA 1 BATCH EXPIRATION UPDATE LISTING

---------------------------------------------------------------------------------------------------------------------------
VOLSER       VOL       FILE     DATA SET NAME                  EXPIRATION     CREATE       CREATE       CREATE      RDS MESSAGE
             SEQ        SEQ                                    DATE           JOBNAME      STEP         DATE
---------------------------------------------------------------------------------------------------------------------------
980005        1          1      CAI.CA1.DATA                   YYYY/072       JOBTAPER     WRITESL      YYYY/034    NO RDS REC
980013        1          1      CAI.CA1.MVMF.DATA              CYCLE/001      JOBTAPER     WRITESL      YYYY/054    NO RDS REC
980016        1          1      CAI.CA1.TAPER.OU01SL01         CYCLE/003      JOBTAPER     WRITESL      YYYY/054    NO RDS REC

END OF REPORT
```

TMSEXPDT Report Field Definitions

PARMS RECEIVED: xxxxxxxx

Parameters passed to program

Control statement image

Input control statement

RETENTION INPUT RECORDS READ

Number of input records read from Retention Data Set (RDS)

RETENTION DEFINITIONS

Number of RDS rules found in Retention Data Set

RETENTION ERRORS FOUND

Number of errors found in Retention Data Set

RETENTION DEFINITIONS IN ERROR

Number of RDS rules found in error in Retention Data Set

Options in effect:

CA 1 options and values used by program

Opening DDNAME EARLOUT: xxxxxxxx

Data set and volume for EARLOUT DD

Records written to EARLOUT

Number of records written to EARLOUT DD

Default volumes selected.

Number of eligible volume records selected from TMC when no RDS rules specified

DSNs eligible, no RDS entry

Number of data sets selected from TMC for processing with no matching RDS rule

Records modified by user exit

Number of TMC records modified by TMSXEXP user exit prior to TMC update

Records bypassed by user exit

Number of TMC records bypassed by TMSXEXP user exit

PARM=xxxxxxxx

Parameters passed to program

VOLSER

Volume serial number of tape volume or file

VOL SEQ

Volume sequence of tape volume

FILE SEQ

File sequence of file

DATA SET NAME

Data set name of tape volume or file

EXPIRATION DATE

New expiration date of tape volume or file

CREATE JOBNAME

Creating job name of tape volume or file

CREATE STEP

Creating step name of tape volume or file

CREATE DATE

Creation date of tape volume or file

RDS EXPDT

Retention Data Set expiration date for tape volume or file

RDS MESSAGE

Processing Messages (see 'TMSEXPDT Messages' in the *Messages Reference Guide*).

TMSFORMT Utility—Format, Extend the TMC and Audit Data Set

TMSXTEND can be used to move the TMC to a new volume, reblock the TMC, and extend the TMC by adding volume ranges or DSNB records. TMSXTEND can also perform these functions however it also has the significant added advantage of being able to be run while tape processing continues. TMSXTEND includes logic to manage the swap-over to a new TMC on all systems sharing the TMC. Refer to the description of the TMSXTEND utility (see page 444) in this guide to learn more about that utility and the functions that it provides that are superior to TMSFORMT.

The TMSFORMT utility must still be used for the TMC maintenance functions listed below if your installation is using non-standard volume serial numbers. TMSFORMT must also still be used to format a new or replace an unusable CA 1 Audit data set.

Note: Once you extend a TMC using TMSXTEND, you must continue to run TMSXTEND, for TMC volume range and DSNB additions or deletions. You cannot run TMSFORMT for anything other than Audit functions after creating a new extended TMC with TMSXTEND.

TMSFORMT is used to perform the following functions:

- Formatting a new TMC

  For new installations of CA 1, TMSFORMT formats the TMC to establish Volume and DSNB records. Initial information regarding the data sets on the volumes may be provided, and ranges of volumes can optionally be marked as inactive (deleted) for future expansion.

- Formatting a new Audit data set or replacing an unusable Audit data set

  TMSFORMT establishes the maximum number of Audit records and the Audit data set block size for new CA 1 installations. In the event the Audit data set is compromised or needs to be expanded, TMSFORMT is used to format a new Audit data set.

- Adding Volume or DSNB records to an existing TMC

  Current CA 1 environments can EXTEND the number of TMC Volume and DSNB records to accommodate library growth. TMSCOPY provides the required input for this procedure.

- Removing Volume records from an existing TMC

  As part of the TMSREMOV procedure, TMSFORMT is used to create a *target* TMC. See the discussion of TMSREMOV in the section TMSREMOV - Removing TMC Volume Records.

- Performing a split/merge operation

  TMSFORMT is executed to (1) create a *target* TMC for an internal split/merge or (2) extend an existing TMC to merge information from another TMC. Internal split/merges are used to reduce the number of DSNB records established. Two batch utilities are used in this process; see the discussion of TMSSPLIT (see page 396) and TMSMERGE (see page 344).

Depending on the TMSFORMT function to be performed, CA 1 may be active, batch active or inactive.

Important! This program should be used only by the personnel responsible for the installation and maintenance of CA 1.

TMSFORMT should **not** be used for the following functions:

- Restoring a TMC or Audit data set

  The space allocated to the TMC does **not** require formatting for the TMSCOPY utility to properly recover the database. The Audit data set cannot be restored. If the Audit data set is compromised, a new one must be formatted. See the discussion on Formatting the Audit Data Set (see page 262).

- Moving a TMC or Audit data set to another DASD unit

  Moving the TMC and Audit data sets to new DASD devices does **not** require that the new allocation be formatted. A complete procedure for moving the databases is documented in the *Programming Guide*.

- Marking a range of tapes in an existing TMC as inactive (deleted)

  Once ranges have been formatted in the TMC and the TMC has been used, ranges should be marked inactive only with the TMSDELET utility.

Note: Documentation regarding DASD space requirements for the TMC and Audit data set can be found in the *Installation Guide*.

Report Description

TMSFORMT produces TMS Report 30 - TMSFORMT Audit Trail. A sample of this report appears on page TMS Report 30 - TMSFORMT Audit Trail (Format) (see page 265).

Formatting a New TMC and Providing Optional Initial Information

See the discussion of TMSXTEND at the start of this section on TMSFORMT as well as the documentation on TMSXTEND later in this guide. TMSXTEND provides a simpler way to extend the TMC, however you must observe the restrictions associated with that utility. Some non-standard alphanumeric volume serial numbers cannot be processed by TMSXTEND.

Formatting the TMC establishes the hashing algorithm used to access the TMC records. It also establishes the values for various control variables in the TMC records. These control variables keep track of the number of Volume records and DSNBs.

When TMSFORMT is executed, a TMC record is built to provide information about each volume under CA 1 control. Data from existing catalogs can be converted to the proper format to provide initial information on the status and content of the volumes. The figure on Job Control Statements (see page 257) shows this process. See the TMSIDATA utility in the *Release Summary* for information on building a TMC from VSAM or system catalog listings.

The TMSCONVR utility can be executed to read a user-supplied data set and format the data into 340-byte records acceptable to TMSFORMT. Each input record must have a volume serial number to identify the TMC record to be updated along with the data set name. The next most critical field is the expiration date because it determines when a volume is available for scratch.

The TMSUPDTE utility can also be used **after** TMSFORMT to set fields such as TRTCH and TMDEN to the correct value when adding new ranges of cartridges that emulate one device type but physically are different.

During the construction of the TMC, you may be aware of future tape library expansion plans, so the volume serial number range or ranges of the new volumes may also be formatted. When TMC records are formatted for tape volumes that do not exist, they should be marked as inactive to suppress their appearance on reports.

Only *new records* or *new record ranges* can be marked as inactive using TMSFORMT (use TMSDELET to mark as inactive records that already exist).

CA strongly recommends that the TMC and Audit data sets be created and maintained as physical sequential unmovable data sets, data set organization of PSU (DSORG=PSU).

CA 1 can control volumes having volume serial numbers that are not six-digit numerics. To accommodate volume serial numbers that are not six digits, two exits have been provided in CA 1 to perform the required numeric translation. The translation scheme is specified at system installation and is used only for internal system processing. Externally, the alphanumeric volume serial number is used to reference the volume.

TMSXITU and TMSXITE, the two user exits which perform the translation, are used during the formatting of the TMC. The alphanumeric exit names are required to be specified on the parameter statement along with PARM=NEW.

- TMSXITU converts a nonnumeric volume serial number into a unique numeric volume serial number that is used *internally* by CA 1 to access the Volume record in the TMC. The version of TMSXITU in CAI.CTAPLINK must be the version that is executed. If a different version is executed through STEPLIB/JOBLIB, an error condition may occur.
- TMSXITE validates the TMSXITU conversion, converts the internal numeric VOLSER back to the installation standard nonnumeric VOLSER, and places the nonnumeric VOLSER into the TMC record so that the nonnumeric VOLSER can be used *externally* to reference the volume.

To test changes to TMSXITU and TMSXITE, STEPLIB to the changed modules and execute TMSFORMT with PARM=NEW against a preallocated TMC. If invalid conversions of TMC records are found, TMSFORMT will issue error messages indicating the discrepancy between the volume translated by TMSXITU and that translated by TMSXITE. Since TMSFORMT issues a LOAD for these two user exits, an IPL is *not* required, just a STEPLIB.

Notes:

- Alphanumeric exit names must be specified on the PARM statement along with a STEPLIB DD to test the alphanumeric exits.
- These exits may be LPA-resident. To use a different version of the exits, specify the appropriate load module library on the STEPLIB DD in the TMSFORMT JCL.

It is important to note that the control statements for TMSFORMT must express the *internal range* for alphanumeric volume serial numbers. For more information on the user exits, see the *Programming Guide*.

Job Control Statements

```
//stepname EXEC
PGM=TMSFORMT[,PARM='[DATEFMT=(fmt)][,DEFAULT][,NEW[,TMSXITE=(your.exite.name),TMS
XITU=(your.exitu.name)]]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TMC      DD  DSN=CAI.CA1.TMC,DISP=OLD,DCB=(BLKSIZE=n*340) *NOTE
//RESTORE  DD  DUMMY
//TMDATA   DD  DSN=tmdata.dsn,DISP=OLD,DCB=(BUFNO=60)
//SYSIN    DD  *
(control statements here)
/*
```

Note: The BLKSIZE may be any multiple of the LRECL, 340, but not 680. When blocking the TMC, the smallest valid block size is 1020. Also, when formatting a small blocked test TMC, ensure that it consists of at least 2 blocks. A recommended value is BLKSIZE=8840. For small block sizes, an appropriate BUFNO=nn value may be added to the DCB parameter to improve performance.

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 30. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

DEFAULT

Indicates that all new records should have the eligible for RDS override bit set in FLAG1 (X'01') to provide retention override processing by TMSEXPDT. This is not needed if TMDATA is DUMMY.

NEW

Indicates that a new TMC is being created, and the CA 1 SVC has not been installed. This will prevent a call to the SVC. NEW must be specified in formatting a test TMC on the production system.

TMSXITE

Specifies the name of the TMSXITE exit. This is required when using PARM=NEW and to support alphanumeric volsers.

TMSXITU

Specifies the name of the TMSXITU exit. This is required when using PARM=NEW and to support alphanumeric volsers.

Notes:

- After the TMC has been formatted with a PARM=NEW,TMSXITE=*your.exite.name*,TMSXITU=*your.exitu.name* update the TMOOPTxx options XITE and XITU to the appropriate user exit name prior to starting TMSINIT.
- Each execution of TMSFORMT without a PARM of NEW creates a special Type 5 Structural Modification Audit record. During TMC RESTORE processing with TMSCOPY, if such a record is encountered, TMSCOPY will abend with a U0010. This may or may not be an abnormal situation; see the discussion on Restoring the TMC with the Current Backup (see page 207) for more information.

JCL Considerations

RESTORE

- Is required and defined as DD DUMMY.

TMDATA

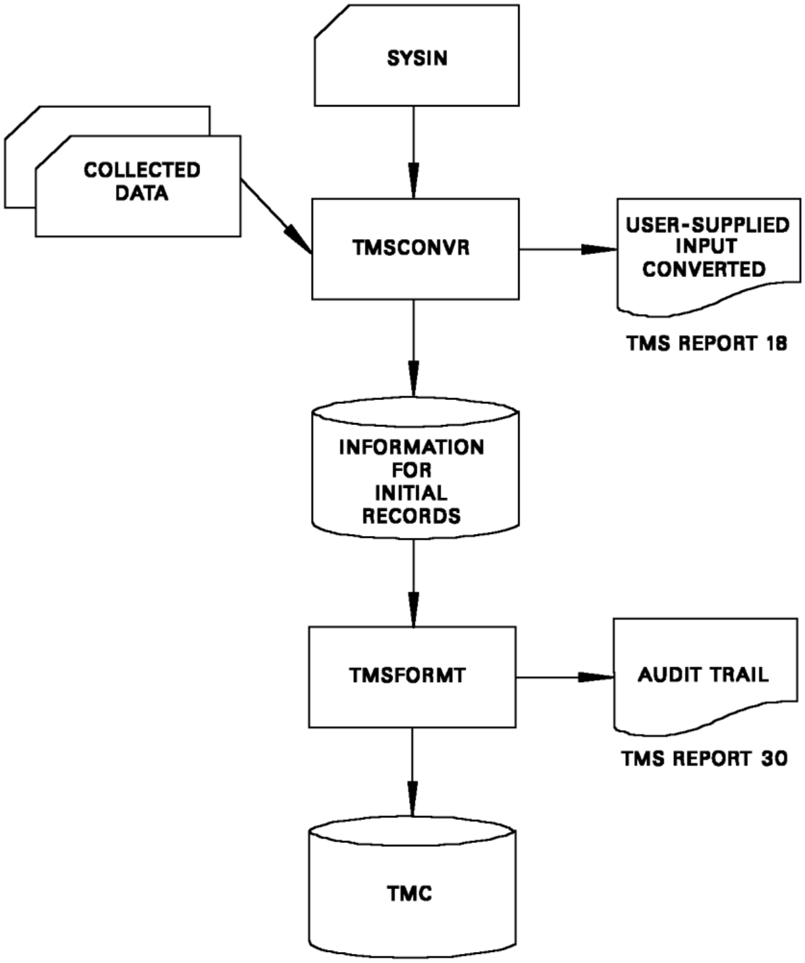
Defines the volume record information output from TMSIDATA or TMSCONVR. This DD statement must be present. If there is no user data, specify:

```
DD DUMMY,DCB=BLKSIZE=340
```

SYSIN

Defines the control statement data set. This may be any sequentially readable data set (LRECL=80).

The following diagram illustrates TMC Creation With TMSFORMT:



Control Statement Specification

Two control statements are used while formatting the TMC: the FORMAT control statement and the DELETE control statement.

FORMAT Control Statement

The FORMAT control statement specifies the TMC volume ranges to be formatted and the number of TMC DSNBs to be formatted. Data on control statements cannot extend past position 71. Multiple FORMAT control statements may be used. This control statement is *required*.

The format for this control statement is:

```
FORMAT TMC=(range1,range2,...,rangen),DSNB=n
```

Control Statement Definition

FORMAT

Identifies the control statement function. FORMAT can begin in any position and must be followed by a space.

TMC=(range1,range2,...,rangen)

Specifies that the ranges indicated by range1 through rangen are to be written on the TMC. The format of rangen is the 1- to 7-digit low-limit *internal* volume serial number followed by the 1- to 7-digit, high-limit *internal* volume serial number. The expression TMC=(rangen) must not be split across two control statements. If more ranges are required than can fit on one control statement, specify additional FORMAT control statements. See the control statement examples for an illustration.

```
TMC=(1000-9999,25000-49999,...,100000-103000)
```

The ranges represent all of the ranges in the new TMC. Only numeric volume serial numbers are valid in this control statement; alphanumeric volume serial numbers must be coded according to their CA 1 *internal* numeric translation.

A maximum of 52 nonconsecutive volume serial ranges can be specified, and each range must be separated by a comma.

DSNB=n

Specifies that n number of DSNBs are to be written on the TMC. The expression DSNB=n must not be split across two control statements. While more than one occurrence of the expression DSNB=n is allowed, the *last* occurrence is the one which TMSFORMT retains. The maximum number of secondary data sets that can be tracked may be determined by the formula found in the *Installation Guide*.

Note: There is a logical limit of 9,999,999 volume records and 6,750,000 DSNB records which can be formatted in the TMC. However, the physical limit is smaller. The limit for the allocation of a sequential data set is X'FFFF' tracks or 65,535 decimal tracks. This is equal to 4,369 cylinders on a 3390. The total number of volume records plus half of the total number of DSNB records (there are two DSNB records on each physical record) must be equal to or less than 3,735,492 for an un-blocked TMC (10,223,460 for a TMC blocked at 8840). Keep in mind, there are 3 control records.

For example, you can format on a TMC blocked at 8840:

- 6,000,000 volume records plus 8,446,914 DSNB records or
- 3,000,000 volume records plus 14,446,914 DSNB records or
- 1,000,000 volume records plus 16,750,000 DSNB records.

DELETE Control Statement

The DELETE control statement tells CA 1 to mark specific TMC records as inactive or deleted while the new ranges are being formatted. Multiple DELETE control statements may be used. This control statement is optional.

The expression mmmmmmm-nnnnnnn cannot be split across two control statements. If more ranges are required than can fit on one control statement, specify additional DELETE control statements. See the control statement examples below for an illustration.

More than one range may be specified, and each range must be separated by a comma. The maximum number of ranges which may be specified is 300.

The format for the DELETE control statement is:

```
DELETE mmmmmmm-nnnnnnn,...,mmmmmmm-nnnnnnn
```

Control Statement Definition

DELETE

Identifies the control statement function. DELETE can begin in any position and must be followed by a space.

mmmmmmm

Is the low-limit internal volume serial number for the range being marked inactive. This value must be a 1- to 7-digit numeric.

nnnnnnn

Is the high-limit internal volume serial number for the range being marked inactive. This value must be a 1- to 7-digit numeric.

Control Statement Examples

These control statements will format a TMC with five ranges of volumes and 5000 DSNBs. The ranges of TMC records beginning with internal volume serial number 200001 through 201000 and 400001 through 401000 are marked inactive.

```
FORMAT TMC=(1-1000,100001-101000,200001-201000)
FORMAT TMC=(300001-301000,400001-401000),DSNB=5000
DELETE 200001-201000
DELETE 400001-401000
```

Formatting the Audit Data Set

Formatting the Audit data set must be done before CA 1 is implemented and anytime the Audit data set is destroyed or needs to be expanded. Formatting establishes the maximum number of Audit records and the Audit data set block size.

There are nine steps to perform when formatting the Audit data set. If it is the initial format, only Steps 5 through 8 are required.

Step 1

Stop tape processing on all CPUs.

Step 2

Back up the TMC using TMSCOPY unless the Audit data set has been destroyed.

Step 3

Execute TMSINIT to *deactivate* CA 1 on *all CPUs*. Respond with the CA 1 system password to the initialization WTOR. Respond to the CTS address space with a command to release the audit data set.

Step 4

Scratch and uncatalog the old Audit data set.

Step 5

Allocate and catalog the new Audit data set (DSORG=PSU).

Step 6

If a new Audit data set name was used, change the parameter in the TMOOPTxx member of *hlq*.CTAPOPTN to reflect the new name.

Step 7

Execute TMSFORMT with the FORMAT AUDIT control statement (see the FORMAT AUDIT Control Statement (see page 264)).

Step 8

Execute TMSINIT on *all* CPUs to reinitialize CA 1. Respond, if necessary, with U to the initialization WTOR.

Step 9

Back up the TMC using TMSCOPY. Respond to the CTS address space with a command to obtain the audit data set.

Steps 1, 4 and 5 are standard system functions. Steps 2, 3, 8 and 9 are CA 1 functions which are explained in the discussions of TMSCOPY and TMSINIT in this guide. Step 6 is explained in the description for the FORMAT AUDIT control statement. For more information on CTS address space commands, see the *Administration Guide*.

Job Control Statements

```
//stepname EXEC PGM=TMSFORMT
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//AUDIT    DD  DSN=CAI.CA1.AUDIT,DISP=OLD,DCB=BLKSIZE=n*370     *NOTE
//RESTORE  DD  DUMMY
//TMDATA   DD  DUMMY,DCB=(BLKSIZE=340,BUFNO=60)
//SYSIN    DD  *
 (control statements here)
/*
```

Notes:

- The BLKSIZE may be any multiple of the LRECL, 370. Formatting of the Audit does fill all of the allocated blocks, which can result in more records being formatted than specified. For example, formatting 16,000 records on a 3380 device, blocked at 8880, will yield 16,008 records in a single extent 10 cylinder allocation.
- When formatting the audit for BLOCKSET mode, there is a minimum BLKSIZE of 4440 and a minimum file size of 100 blocks. One additional block for the control records is added automatically.

JCL Considerations

AUDIT DD

Defines the new Audit data set.

RESTORE DD

Must be present and must be specified as DD DUMMY.

TMDATA

Must be present and must be specified as:

```
DD DUMMY,DCB=BLKSIZE=340
```

SYSIN DD

Defines the control statement data set. This may be any sequentially readable data set (LRECL=80) or may be specified in-stream.

Control Statement Specification

The FORMAT AUDIT control statement is used to format the Audit data set.

FORMAT AUDIT Control Statement

```
1       8      14

FORMAT AUDIT=nnnnnnn[,BLOCKSET]
```

Control Statement Definition

FORMAT

Specifies the control statement function. FORMAT can begin in any position and must be followed by a space.

AUDIT=nnnnnnn

Specifies that the Audit data set is to be formatted. The value of nnnnnnn is the total number of records to be formatted in the Audit data set. This number should be based on the amount of DASD space allocated. See the *Installation Guide* for instructions on calculating this number.

Note: The THAUDT parameter in *hlq*.CTAPOPTN specifies the percentage of Audit records which can be used before warning messages for backup are issued. The default value for THAUDT is 80 percent.

BLOCKSET

Specifies that the Audit data set is to be formatted for the BLOCKSET mode. The audit data set formats in complete blocks. Control record 4 and a new control records 5 occupy the first block. The first available audit record starts in the second block. The number of records specified by AUDIT=nnnnnn is round up to a complete block. An extra block of records is added to contain the control records.

Note: CA 1 programs will recognize this format and process the audit data set in "BLOCKSET mode".

Example:

This control statement example will format the Audit data set with 1000 records.

```
FORMAT AUDIT=1000
```

If the Audit threshold is set at 85 percent in hlq.CTAPOPTN, after 850 records are used, the operator is prompted with the IEFTMS9, AUDIT FILE UTILIZATION IS CURRENTLY nn% message for each percentage of utilization until the condition clears.

```
FORMAT AUDIT=1000,BLOCKSET
```

If the Audit data set was allocated with a BLKSIZE=4440, it would be formatted for BLOCKSET mode and contain 1020 records. A BLKSIZE of 4440 can contain 12 audit records so the 1000 records requested would be rounded up to 1020 to make all blocks 4440 bytes long. Then an additional block would be added to contain control records 5 and 4.

TMS Report 30 - TMSFORMT Audit Trail (Format)

```
SATURDAY, JUNE 22, 2005.174                      Y O U R   C O M P A N Y   N A M E                             ROxxxxx PAGE=00001

TMS REPORT 30 - TMSFORMT AUDIT TRAIL

FORMAT TMC=(000000-000999,100000-104999,500000-504999)                                  1
FORMAT TMC=(600000-600999),DSNB=25000                                                   2
FORMAT AUDIT=25000                                                                      3

NUMBER OF TMC RECORDS WRITTEN.                      12000
NUMBER OF RANGES FORMATTED ON THE TMC.                  4
TOTAL NUMBER OF DSNB'S ON THE TMC.                  25000

NUMBER OF AUDIT RECORDS WRITTEN.                    25000
NUMBER OF AUDIT RECORDS PER BLOCK.                     10

TMS FORMAT PROGRAM SUCCESSFULLY COMPLETE.
```

Extend the TMC

Before extending the TMC, verify that sufficient space is available. The IBM IEHLIST or IDCAMS utilities or ISPF Version 3.2 or 3.4 functions may be used to determine the amount of unused space. The *Installation Guide* may be used to help calculate the space required.

It is strongly recommended, but not required, that TMSPTRS be executed to detect any chaining errors that should be corrected in the TMC. If any of the volumes being added have volume serial numbers that are not six-digit numerics, two user exits are required to perform the alphanumeric to numeric translation described on Formatting a New TMC and Providing Optional Initial Information (see page 255). For more information on user exits TMSXITU and TMSXITE, see the *Programming Guide*.

CA 1 must be batch active to extend the TMC. During the extend operation, data can be entered to build Volume records for the new volumes or ranges of volumes being added. Deviations from the following procedure may result in an incorrect or unusable TMC.

If you are planning on extending the TMC with volsers that are alphanumeric then you will need to update your conversion exits, TMSXITE and TMSXITU. This can be done with TMSXITE and TMSXITU or with TMSVOLDF.

Step 1

Stop tape processing on all CPUs.

Step 2

Back up the TMC using TMSCOPY.

Step 3

Batch activate CA 1 on the CPU running TMSFORMT using TMSINIT. Respond to the IEFTMS0 initialization prompt with system.password,BATCH. For all other CPUs sharing the TMC, place CA 1 in an inactive state using TMSINIT. Respond to the IEFTMS0 initialization prompt with system.password. Respond to the CTS address space with a command to RELEASE the TMC.

Note: In a batch active state, such facilities as CA 1 ISPF, TIQ and TMSUPDTE perform relative record updates and do not go through an OPEN. Therefore, make sure that none of these facilities are in use at the time of the TMSFORMT extend process.

Step 4

Allocate a new TMC unless the original allocation is being used.

```
DCB=(RECFM=FB,LRECL=340,BLKSIZE=n*340,DSORG=PSU,BUFNO=60)
```

Note: The BLKSIZE may be any multiple of the LRECL, 340, but not 680. When blocking the TMC, the smallest valid block size is 1020. Also, when formatting a small blocked test TMC, make sure that it consists of at least 2 blocks. A recommended value is BLKSIZE=8840. For small blocksizes, an appropriate BUFNO= value may be added to the DCB parameter to improve performance.

Step 5

Execute TMSFORMT with the EXTEND control statement using the backup tape from Step 2 as input to the RESTORE DD. If you are formatting an alphanumeric range you will need to add a STEPLIB to the JCL pointing to the new TMSXITE and TMSXITU modules.

Step 6

If the extend fails because there is insufficient space (SD37 ABEND), allocate a larger new TMC and return to Step 5.

Step 7

If a new TMC data set name was used, make sure that it is cataloged to all CPUs, and that the TMC data set name parameter (TMCDSN) in the TMOOPTxx member of *hlq*.CTAPOPTN reflects the same name.

Step 8

If you formatted the TMC using new exits, they must be loaded into LPA through CAS9 before you start TMSINIT. Change the CARIMPRM control statement in CAS9 for CA 1 to:

```
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)          -
PARM(REINIT,LPA=TMSXITE)                                                -
LOADLIB(your load library)
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)          -
PARM(REINIT,LPA=TMSXITU)                                                -
LOADLIB(your load library)
```

You need 2 statements so that both exits are loaded. Execute TMSINIT on *all* CPUs to reactivate CA 1.

If using TMSVOLDF change the CARIMPRM control statement in CAS9 for CA 1 to:

```
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)         -
PARM(REINIT,LPA=TMSVOLDF)                               -
LOADLIB(your load library)
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)         -
PARM(REINIT,LPA=TMSXITE)                               -
LOADLIB(your load library)
PRODUCT(CA 1/MVS) VERSION(L0C6) INIT(L0C6INIT)         -
PARM(REINIT,LPA=TMSXITU)                               -
LOADLIB(your load library)
```

Step 9

Back up the new TMC using TMSCOPY. Respond to the CTS address space with a command to obtain the TMC.

Note: In a batch active state, facilities as CA 1 ISPF, TIQ and TMSUPDTE perform relative record updates and do not go through an OPEN. Therefore, care must be taken to ensure that none of these facilities are in use at the time of the TMSFORMT EXTEND process.

Steps 1, 4, 6 and 7 are standard system functions. Steps 2, 3, 8 and 9 are CA 1 functions which are explained in this section of the manual under TMSCOPY and TMSINIT. Step 5 is explained in this subsection under the EXTEND control statement.

Note: For more information on CTS address space commands, see the *Administration Guide*.

If all attempts to extend the TMC fail when the original allocation is used, then restore the TMC with TMSCOPY using a PARM of RESTORE, NOAUDIT. Use the TMC backup from Step 2 and then back up the restored TMC with TMSCOPY.

If all attempts to extend the TMC fail when the original allocation is *not* used, recatalog the old TMC on all CPUs.

Extending the TMC is a critical process. Immediately after an extend, verify that the TMC has been extended as desired by using CA 1 ISPF or TIQ, TMSBINQ (HEADERS) and CA Earl reports. Also, verify that all CPUs are using the same TMC. If the extended TMC is not as desired, follow the instructions in the previous two paragraphs for restoring or recataloging the original TMC.

The extend procedure may be tested by creating a new TMC, using a different data set name than the production TMC and performing Steps 1 through 6. DO NOT use the existing allocation for the production TMC. Step 8 may then be performed on the CPU executing the extend. You can reference the test TMC by either (1) changing the TMCDSN option in the TMOOPTxx member of *hlq*.CTAPOPTN, or by (2) renaming the production TMC to a backup name and the test TMC to the production name.

You can verify that the extend operation was successful by using the CA 1 batch or online facilities mentioned above. To reestablish the production TMC, either (1) make sure the TMOOPTxx reference is correct, or (2) rename the production TMC to its proper name and the test TMC to a different name.

Console messages are issued during TMSFORMT EXTEND processing. These messages are documented in the *Messages Reference Guide*.

Job Control Statements

```
//stepname EXEC PGM=TMSFORMT[,PARM=DEFAULT]
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMC      DD  DSN=CAI.CA1.TMC,DISP=OLD                 *NOTE
//RESTORE  DD  DSN=backup.dsn,DISP=OLD,DCB=(BUFNO=60)
//TMDATA   DD  DSN=tmdata.dsn,DISP=OLD,DCB=(BUFNO=60)
//TMSRPT   DD  SYSOUT=A
//SYSIN    DD  *
(control statements here)
/*
```

Note: For small TMC block sizes, an appropriate DCB=BUFNO=nn value may be added to the DCB parameter to improve performance.

Parameter Definitions

DEFAULT

Indicates that all new records are to have the eligible for RDS override bit set to allow for processing by the Retention Data Set. This parameter is not needed if there is no TMDATA input.

JCL Considerations

TMC

Defines the new TMC.

RESTORE

Defines the latest copy of the TMC backup.

TMDATA

Defines the user data output from TMSIDATA or TMSCONVR. This DD statement must be present. If there is no user data, specify:

```
DD  DUMMY,DCB=BLKSIZE=340
```

SYSIN

Defines the control statement data set. This may be any sequentially readable data set (LRECL=80).

STEPLIB

Defines the library containing the user exits to convert alphanumeric volsers (TMSXITE and TMSXITU). If this DD is not specified, these user exits are used from the link list. TMSFORMT does not use the LPA resident versions of TMSXITE and TMSXITU.

Control Statement Specification

Two control statements are available for an extend: EXTEND and DELETE.

EXTEND Control Statement

The EXTEND control statement specifies the *new* TMC volume ranges to be formatted or the additional number of DSNB records to be added. Multiple EXTEND control statements may be used. This control statement is *required*. The format for the EXTEND control statement is:

```
EXTEND TMC=(range1,range2,...,rangen),DSNB=n
```

Control Statement Definition

EXTEND

Identifies the control statement. EXTEND can begin in any position and must be followed by a space.

DSNB=n

Represents the number of DSNBs to be *added* to the current TMC. The expression DSNB=n must not be split across two control statements. While more than one occurrence of DSNB=n is allowed, the last occurrence is the one TMSFORMT retains.

TMC=(range1,range2,...,rangen)

Specifies that the new ranges indicated by range1 through rangen are to be added to the current TMC. The format of rangen is the 1- to 7-digit, low-limit *internal* volume serial number followed by the 1- to 7-digit, high-limit *internal* volume serial number. Old ranges must not be specified.

```
EXTEND  TMC=(1000-9999,25000-49999,...,100000-103000)
```

The expression TMC=(rangen) must not be split across two control statements. If more ranges are required than can fit on one control statement, specify additional EXTEND control statements. The total number of nonconsecutive ranges (previously existing ranges and new ranges) cannot exceed 52. Previous and new ranges that are internally consecutive are consolidated into one range.

Control Statement Example

This control statement adds a new range to the TMC, internal volume serial numbers 2051 through 3000, and adds 50 DSNBs:

```
EXTEND  TMC=(2051-3000),DSNB=50
```

DELETE Control Statement

The DELETE control statement tells CA 1 to mark specific TMC records as inactive or deleted while the new ranges are being formatted. Multiple DELETE control statements may be used. This control statement is optional.

The expression mmmmmmm-nnnnnnn cannot be split across two control statements. If more ranges are required than can fit on one control statement, specify additional DELETE control statements. See the control statement examples below for an illustration.

More than one range may be specified, and each range must be separated by a comma. The maximum number of ranges which may be specified is 300.

The format for the DELETE control statement is:

```
DELETE mmmmmmm-nnnnnnn,...,mmmmmmm-nnnnnnn
```

Control Statement Definition

DELETE

Identifies the control statement function. DELETE can begin in any position and must be followed by a space.

mmmmmmm

Is the low-limit internal volume serial number for the range being marked inactive. This value must be a 1- to 7-digit numeric.

nnnnnnn

Is the high-limit internal volume serial number for the range being marked inactive. This value must be a 1- to 7-digit numeric.

TMS Report 30 - TMSFORMT Audit Trail (Extend/Delete)

```
FRIDAY, MARCH 01, 2002.060                Y O U R C O M P A N Y   N A M E  GmbH                               ROxxxxx  PAGE=00001
TMS REPORT 30 - TMSFORMT AUDIT TRAIL

FORMAT TMC=(000000-000999,100000-100299,100600-100699)                                1
FORMAT TMC=(100800-100999,144000-145299,186900-187399)                                2
FORMAT TMC=(205100-205199,330500-330999),DSNB=10000                                   3
FORMAT AUDIT=25000                                                                    4

NUMBER OF TMC RECORDS WRITTEN.                        4000
NUMBER OF RECORDS PER BLOCK.                            26
NUMBER OF RANGES FORMATTED ON THE TMC.                   8
TOTAL NUMER OF DSNB'S ON THE TMC.                    10000
NUMBER OF AUDIT RECORDS WRITTEN.                     25008
NUMBER OF AUDIT RECORDS PER BLOCK.                      24

TMS FORMAT PROGRAM SUCCESSFULLY COMPLETE.
```

BLOCKSET Mode

The BLOCKSET mode is a performance feature for customers with large and active audit data sets. In this mode, records in the audit data set are allocated by blocks instead of records. Each allocated block is then assigned to an LPAR for its exclusive use. The assignment greatly reduces the number of RESERVEs and updates of the audit control record 4. It is not necessary to read a block before adding a record. The block is written each time that a record is added but the buffer is retained by the LPAR. It is not necessary to reread the block. After the LPAR has written all the records in the block, the LPAR will allocate another block with a RESERVE and update of control record 4. When an LPAR allocates a block, the next available record pointer in control record 4 is incremented by the number of records in a block.

The new control record 5 is the first record in a BLOCKSET audit data set. Control record 4 is now the second record. The remainder of the first block is padded with dummy audit exception records. Control record 5 contains a table of up to 32 LPARs, each LPAR with the record number of the block that is allocated to it. Control record 5 also contains a timestamp of when a block was last allocated.

Each time a block is allocated, a check is made for the audit threshold and full conditions. When the physical end of the audit data set is reached, it will wrap around to the second block in the data set. Only control records occupy the first block. When calculating percent used, all the records in a block are consider used. The blocks for the control records are not counted.

Universal Audit Time

When a BLOCKSET audit file is accessed, the file enables the Universal Audit Time feature. The feature provides greater precision and support for multiple time zones. All audit time stamps in both the TMC and audit are processed as GMT/UTC date and time. The feature provides a single time reference for all LPARs in any time zone. The precision of the time field is also increased from HHMM to HHMMSSth and is recorded in a 4 byte packed unsigned field. The record allows accurate analysis of TMC updates from active LPARs in one or more time zones.

TMSFORMT Report Field Definitions

Control statement image

Input control statement

NUMBER OF TMC RECORDS WRITTEN.

Number of Volume records written

NUMBER OF TMC RECORDS PER BLOCK.

Number of TMC records in a block

NUMBER OF RANGES FORMATTED ON THE TMC.

Number of formatted ranges on the TMC

TOTAL NUMBER OF DSNB'S ON THE TMC.

Number of DSNB records formatted

NUMBER OF AUDIT RECORDS WRITTEN.

Number of Audit records formatted

NUMBER OF AUDIT RECORDS PER BLOCK.

Number of Audit records in a block

TMSFVSN Utility—Initialize DSNBs with Beginning Volume for Data Set

Use the TMSFVSN utility to initialize the DSNB field DSNBFVSN in all used DSNB records.

This field is used only for the security system. CA recommends that you run TMSPTRS before running TMSFVSN.

CA 1 must be active or batch active to run TMSFVSN.

Important! No CA 1 routines or tape activity should occur during execution of TMSFVSN.

Report Description

TMSFVSN produces TMS Report 59 - DSNB Initialization Report which lists errors and statistics. A sample of this report appears on TMS Report 59 - DSNB Initialization Report (see page 274).

Job Control Statements

```
//TMSFVSN  EXEC PGM=TMSFVSN[,PARM='[TEST][,LIST]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//WORKFILE DD  UNIT=SYSDA,SPACE=(CYL,(10,2))
//SYSOUT   DD  SYSOUT=A
//SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK06 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
```

Note: The number of Audit records required when this program is executed is one for each used DSNB. If a U0008 ABEND occurs because of too few Audit records, rerun this program after doing a TMSCOPY BACKUP. Only those records not previously updated are updated.

Parameter Definitions

TEST

Indicates that no updates are performed.

LIST

Produces a detailed list of updated DSNBs.

Control Statement Specification

There are no control statements for TMSFVSN.

TMS Report 59 - DSNB Initialization Report

```
FRIDAY, AUGUST 05, 2005.215                 Y O U R   C O M P A N Y   N A M E                        ROxxxxx PAGE=00001

TMS REPORT-59                                                 INITIALIZE DSNB FIELD DSNBFVSN


TMSFVSN-07 00000004 DSNBS UPDATED
TMSFVSN-08 00000000 DSNBS PREVIOUSLY UPDATED
TMSFVSN-09 00000000 DSNBS UPDATED BUT HAD UNKNOWN DATA IN RECORD
```

TMSFVSN Report Field Definitions

DSNBS UPDATED

Number of DSNB records updated

DSNBS PREVIOUSLY UPDATED

Number of DSNB records already updated

DSNBS UPDATED BUT HAD UNKNOWN DATA IN RECORD

Number of DSNB records updated that contained invalid data

TMSGRW Utility—Produce Customized Reports on TMC and Audit Records

Use the TMSGRW generalized report writer to select TMC or Audit records and print reports in a customized format.

Record selection is based on your site-defined values, using relational expressions (such as less than, greater than, or equal to) or Boolean logic operating on input data set fields or your defined constants (literals).

CA 1 can be active, batch active, or inactive. CA Earl supersedes TMSGRW functions. Do all reporting with CA Earl.

The support for TMSGRW is continued. The current functionality of TMSGRW is stabilized while any new functionality is added to the CA Earl and/or CA Easytrieve interfaces to CA 1.

Note: When the audit file is in BLOCKSET mode, all time stamps in either the TMC or audit file are in GMT/UTC.

Input Data Set

The input data set must contain TMC or Audit records. Valid data sets are:

- the TMC
- the Audit, for Audit record types 0, 1, 2, 3, and 4
- the TMC formatted data set output from TMSCONVR
- the Audit backup data set, for Audit record types 0, 1, 2, 3 and 4
- output from a previous TMSGRW execution

The input DD statement can be any data set name. If the input ddname in the TMSGRW control statements is TMC, no DD statement needs to be included in the TMSGRW JCL when CA 1 is active or batch active because the TMC data set is dynamically allocated.

Record Selection

Record selection is based on comparisons of a TMC/Audit record field to a site-defined constant or to another TMC/Audit record field. The comparisons are any combinations of relational expressions.

Example

```
CONTROL-SECTION
       .
       .
       .
      DEFINE CYCLOW DT 'CYCLE/001'
      DEFINE CYCHI DT 'CYCLE/364'
PROCESS-SECTION
      WHEN EXPDT GE CYCLOW AND EXPDT LE CYCHI
      LIST
REPORT-SECTION
       .
       .
       .
```

In the previous example, records for all tape data sets with expiration dates under Cycle Control would be selected.

If a record fails to be selected by a WHEN statement, it is processed against the next WHEN statement. If the TMC/Audit record fails to be selected by any of the WHEN statements, it will not appear in the report.

Complete instructions on record selection are found in the paragraphs covering the PROCESS-SECTION and CONTROL-SECTION statements.

User-Defined Output

You are able to define the contents and format of the TMSGRW report. Through the use of control statements, you can:

- Specify the sort sequence of the report.
- Define the width and length of the printed report page.
- Specify the report title or page heading. The report title is centered and printed at the top of each page.
- Specify a footer line. This line is printed at the end of the report.
- Define the TMC/Audit fields to be printed for each record.
- Specify the sequence of the printed fields across the page.
- Specify column headings for the printed fields.
- Request a top-of-form if the value of the major sort field changes.

Complete instructions on defining the report output are found in the paragraphs covering the REPORT-SECTION. At the end of the report, the number of input records read (into the sort, if sorted) and the number of records selected for output are printed.

The TMS logo (banner) pages preceding the report will reflect the date of program execution. Any other date must be specifically defined in the report title, for example, DDMMM YYYY. Suppressing TMS Banner Pages (see page 30) explains how to suppress the printing of TMS banner pages. You can also suppress the TMC FLAG Bit Definition Page, if desired. See Suppressing TMS Banner Pages (see page 30) for instructions.

The TMC/Audit fields requested must fit into the specified width of the page, up to a maximum of 132 characters. Each field or column is separated by three spaces. The width of each column equals the greatest of: the length of column heading 1, the length of column heading 2, or the maximum length of the TMC/Audit field in character (EBCDIC) format. For example, the width of the following column is eight positions. The first column heading, CREATION, has the longest length.

```
CREATION
UNIT
xxx
```

Unique TMSGRW Keywords

The following keywords may be used only with TMSGRW. The TMC and Audit keywords listed inthe *Programming Guide* are also available for use.

I-TIME

Current time expressed as hhmm. Format is PD (packed decimal).

I-DATE

Current date expressed as ccyyddd. Format is DT (date).

Y-DATE

Yesterday's date expressed as ccyyddd. Format is DT (date).

T-DATE

Tomorrow's date expressed as ccyyydd. Format is DT (date).

Job Control Statements

```
 //stepname EXEC PGM=TMSGRW[,PARM='[DATEFMT=(fmt),][NOLIST,]SCAN=xx'][,REGION=1M]
 //STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
 //TMSRPT   DD  SYSOUT=A
[//anyname1 DD  DSN=dsname,DISP=SHR                       ]
[//anyname2 DD  DSN=include.dsn,DISP=SHR                 ]
[//OUTPUT   DD  DSN=tape.output,DISP=(NEW,KEEP),UNIT=TAPE]
 //SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
 //SYSOUT   DD  SYSOUT=A
 //SORTWK01 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SORTWK02 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SORTWK03 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SORTWK04 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SORTWK05 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SORTWK06 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
 //SYSIN    DD  *
 (control statements here)
 /*
```

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on reports generated by TMSGRW. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

SCAN={*72*|xx}

Is the number of positions that are to be scanned when reading the data pointed to by the SYSIN DD. The default is 72.

NOLIST

Specifies that the control statements are not to be listed when TMSGRW is executed.

JCL Considerations

anyname1

Is any site-defined ddname. This DD statement is optional and defines the data set from which records are selected to produce the report. If TMC is specified, CA 1 macros are used to read sequentially through the TMC. In addition, READ access to the TMC is required. Any other ddname will result in QSAM access of the data set. See the ALTER control statement (see page 283) information in CONTROL-SECTION in the following topics.

anyname2

Is the ddname of a partitioned data set pointed to by an INCLUDE control statement. There may be as many DD statements of this type as there are INCLUDEs, which require different data sets. This DD statement is optional.

OUTPUT DD

Defines a sequential file that contains all TMC records selected by PROCESS-SECTION logic. The LRECL is the same as the input file. Records are sorted in the same order as the report. This DD statement is optional.

Control Statement Specification

There are three classes of control statements:

- CONTROL-SECTION
- PROCESS-SECTION
- REPORT-SECTION

For each class of control statement, there is a separate section. Within each section there are several types of control statements, which are described under the section in which they appear. In addition, one control statement, the INCLUDE control statement, may appear in any one of the three sections. The INCLUDE control statement is discussed first, before the control statements unique to each separate section.

Each execution of TMSGRW requires at least one control statement for each section. The sections must appear in the order shown.

You may include comment statements by placing an asterisk (*) in position 1. Text for a comment may appear anywhere between positions 2 and 80 inclusive.

Examples of TMSGRW control statements and the reports they produce are provided after the description of the various control statements.

INCLUDE Control Statement

The INCLUDE control statement, which may be submitted by itself or in any one of the three sections, specifies a member of a Partitioned Data Set (PDS) which contains TMSGRW control statements. The data set name for the PDS member is specified with the anyname2 DD statement. The PDS member can specify control statements for a partial section or for multiple sections, and *must not* contain an INCLUDE control statement.

The block size of the PDS member will affect the total number of DEFINE statements, WHEN/AND/OR statements and PRINT/DUMP/TRANSLATE statements that can be specified in the PDS member. See Special Considerations (see page 295) for additional information. The format for the INCLUDE statement is:

```
INCLUDE anyname2 xxxxxxxx
```

Control Statement Definition

INCLUDE

Identifies the control statement. It must be followed by a space.

anyname2

Identifies the PDS containing the member that is used as input. anyname2 is the ddname of the PDS as defined in the JCL.

xxxxxxxx

Identifies the PDS member containing TMSGRW control statements.

Examples:

To include PDS member XYZ containing the complete set of control statements for the CONTROL-SECTION, PROCESS-SECTION and REPORT-SECTION, the following SYSIN control statement data set could be used:

SYSIN Data Set	PDS Member XYZ

`INCLUDE PDS XYZ`	`CONTROL-SECTION`
	`(control statements)`
	`PROCESS-SECTION`
	`(control statements)`
	`REPORT-SECTION`
	`(control statements)`

To include PDS member JKL containing control statements for the PROCESS-SECTION, the following SYSIN control statement data set could be used:

SYSIN Data Set   PDS Member JKL

```
CONTROL-SECTION         WHEN EXPDT LE TODAY
(control statements)    REJECT
PROCESS-SECTION
INCLUDE PDS JKL
REPORT-SECTION
(control statements)
```

Control Statements for the CONTROL-SECTION

The control statements available for use in the CONTROL-SECTION are:

- The CONTROL-SECTION control statement identifies the start of the CONTROL-SECTION. This control statement is *required*.
- The SORT control statement defines the sort sequence that the data is to be sorted into for the report. This control statement is optional.
- The NOSORT control statement suppresses all sorting. This control statement is optional. Records are processed in the order the VOLSERs were formatted on the TMC (normally VOLSER ascending).
- The TITLE control statement defines the title of the report. This control statement is optional.
- The FOOTER control statement defines a statement to be printed on the last page of the report. This control statement is optional.
- The ALTER control statement allows the user to modify program defaults. For example, the number of lines per page can be changed. This control statement is optional.
- The DEFINE control statement allows the user to define literals to be used in the comparisons in the PROCESS-SECTION. This control statement is optional, but normally used.
- The INCLUDE control statement allows the inclusion of control statements from a PDS member. This control statement is optional.

CONTROL-SECTION Control Statement

The CONTROL-SECTION control statement identifies the start of the CONTROL-SECTION. This control statement is *required*. Only one CONTROL-SECTION control statement may be submitted per program execution. The format for the CONTROL-SECTION control statement is:

```
CONTROL-SECTION
```

Control Statement Definition

CONTROL-SECTION

Identifies the control statement. It must be followed by a space.

SORT Control Statement

The SORT control statement defines the sort sequence by which the input data is sorted for the report. Only one SORT control statement is allowed. This control statement is optional. However, it *must* be submitted if the NEWPAGE control statement in the REPORT-SECTION is used. If the SORT control statement is not submitted, the input file is sorted into ascending sequence by volume serial number. The format for the SORT control statement is:

```
SORT [sequence-]keyword... [sequence-]keyword
```

Control Statement Definition

SORT

Identifies the control statement. It must be followed by a space.

sequence

Defines the sequence order by which the data is sorted. Valid sequences are:

A

Ascending order (default)

D

Descending order

The sequence must be separated from the following keyword by a dash (-). If sequence is not specified, the default sequence is A (ascending order).

keyword

Defines the field to be sorted.

Multiple SORT fields (sequence-keyword) may be specified by separating the expressions by a space. The first keyword specified is the major sort field. There is a limit of six sort fields.

NOSORT Control Statement

The NOSORT control statement suppresses any requested or default sorting. If NOSORT is specified, the report is in the same order as the input file. The format for the NOSORT control statement is:

```
NOSORT
```

Control Statement Definition

NOSORT

Identifies the control statement. It must be followed by a space.

TITLE Control Statement

The TITLE control statement defines the title of the report. The statement enclosed in single quotes is centered, based on 132 characters per line, at the top of each new page of the report. Only one TITLE control statement is allowed. The format for the TITLE control statement is:

```
TITLE 'report-title'
```

Control Statement Definition

TITLE

Identifies the control statement. It must be followed by a space.

report-title

The report-title is centered, based on 132 characters per line, and printed at the top of each new page of the report. The report-title must be enclosed in single quotes. Embedded quotes will cause the title to be truncated.

FOOTER Control Statement

The FOOTER control statement defines a statement to be printed on the last page of the report. Only one FOOTER control statement is allowed. The format for the FOOTER control statement is:

```
FOOTER 'statement'
```

Control Statement Definition

FOOTER

Identifies the control statement. It must be followed by a space.

'statement'

The 'statement' is centered, based on 132 characters per line, and printed on the last page of the report. The 'statement' must be enclosed within single quotes. Embedded quotes will cause the footer to be truncated.

ALTER Control Statement

The ALTER control statement allows you to change program defaults. Multiple ALTER control statements may be specified. The format for the ALTER control statement is:

```
          [           {nnXppp     }]  [     {NO  }]
ALTER     [PRINTIMAGE={defaultX132}]  [DSNB={YES }]
                                      [     {FAST}]
[      {anyname1}]          [    {SCAN  }]
          [INPUT={TMC     }]          [RUN={TOTALS}]
```

Control Statement Definition

ALTER

Identifies the control statement. It must be followed by a space.

PRINTIMAGE={nnXppp|*defaultX132*}

The value of nn defines the number of lines per page, including titles, and must be two digits. The TMSPRINT LC (line count) default is 55 lines per page. The value of ppp defines the character width of the paper. If specified, ppp must be three digits, the value of which is less than or equal to 132. If the lines per page or the print positions are to be changed, both values must be specified. For example: PRINTIMAGE=40X132.

DSNB={*NO*|YES|FAST}

Specifies whether used DSNBs for the secondary files are to be merged in at read time. The default is NO.

If DSNB=YES is specified, for each used DSNB record a YSVC call is made to retrieve volume information to be combined with the DSNB information. The DSNB fields are accessed by specifying the TMC equivalent keyword. The F1STVSN value is mapped into the VOLSER field so that reports will display the VOLSER on which the data set actually begins. Fields such as OUTCODE, OUTDATE and ACCT are available for the DSNB data sets.

DSNB=YES can be used only with the production TMC. If an ALTER INPUT= is found indicating another file, the logic will change to DSNB=FAST.

Note: DSNB=YES will have an increase in processing time due to the YSVC call. DSNB field not reported is PREV.

If DSNB=FAST is specified, each DSNB field is moved into the corresponding TMC record field. If there is no corresponding TMC record field, an asterisk (*) is moved to that field on the report. The DSNB fields are accessed by specifying the TMC equivalent keyword. NUMDSNB is changed to represent the file sequence number, not the number of secondary data sets. 1STDSNB is changed to represent the DSNB number in the NEXT field. FLAG1 represents the FLAG1 of the DSNB (see the *Programming Guide* for the definition of the flag bits). If the NUMDSNB keyword, which indicates the file sequence number, is not used to distinguish between volume records and DSNBs, it is assumed that no record specific flag handling is desired, and only the corresponding flag bits are moved from the DSNB FLAG1 into the volume FLAG1. The corresponding bits are:

X'80'	TMINTAL	DSNBUSRU	Rec updated by user
X'10'	TMABEND	DSNBABND	File closed by abend
X'01'	TMDFAULT	DSNBDFLT	RDS override enabled

If the NUMDSNB keyword is found on any WHEN statement, the entire DSNB FLAG1 is moved unchanged.

Note: Fields not copied from DSNB records using DSNB=FAST are F1STVSN and PREV.

INPUT=anyname1

Specifies the input ddname. This expression is only required if anyname1 is not TMC or CA 1 is not active. If neither INPUT= or anyname1 DD statement is included, the active TMC is dynamically allocated and used as input.

RUN={SCAN|TOTALS}

Modifies the processing logic of TMSGRW.

SCAN

Performs only a syntax scan of the control statements.

TOTALS

Allows the program to produce only the total number of records selected without producing a detail report. This option supersedes any instruction in the REPORT-SECTION.

DEFINE Control Statement

The DEFINE control statement allows you to set up values (literals) to be used in the comparisons specified in the PROCESS-SECTION. Your keywords are assigned the character, packed decimal, binary, hexadecimal or date values specified. Multiple DEFINE control statements may be specified.

The total value of all DEFINE constants must be less than 14,984 bytes, which allows 340 44-position character fields or 3,746 7-position packed fields (a 7-position packed field uses only 4 bytes of storage). The format for the DEFINE control statement is:

```
DEFINE keyword format 'value'
```

Control Statement Definition

DEFINE

Identifies the control statement. It must be followed by a space.

keyword

Is your defined keyword name that is used in comparisons in the PROCESS-SECTION. The keyword must not be a CA 1 keyword or special TMSGRW keyword. The maximum length of the keyword name is seven characters. It must be followed by a space.

format

Defines the format of the value. Format is two characters and must be followed by a space. Valid formats are:

BI

- Binary

CH

- Character

DT

- Date

HX

- Hexadecimal

PD

- Packed Decimal

The format of your keyword typically must match the format of the field it is being compared against. For example, to compare your keyword against the TMC expiration date, the format of your keyword must be defined as DT (date). It is also possible to use hexadecimal comparison against any field. For example, if you wanted to determine which DSN fields were storing hex zeros, you could define your keyword as: keyword HX '00'

'value'

Defines the actual value to be assigned to your defined keyword. The value must be delimited by single quotes or any other character having a value less than character A. The beginning and ending delimiter must be the same character. Example values are:

BI

- '5000'

CH

- 'AC$TMS$Z'

DT

- 'DDMMMYYYY' (must be expressed in the date format currently selected or in the yyyy/ddd Julian format)

HX

- 'C8F1' (must be even number of hex digits)

PD

- '88002'

INCLUDE Control Statement

The INCLUDE control statement is discussed earlier in this section.

Control Statements for the PROCESS-SECTION

The control statements available for use in the PROCESS-SECTION are:

- The PROCESS-SECTION control statement identifies the start of the PROCESS-SECTION. This control statement is *required*.
- The WHEN control statement defines the test used to determine if the TMC or Audit record is to be processed by the ACTION control statement. Every WHEN control statement must be immediately followed by an ACTION control statement.
- The ACTION control statement (LIST or REJECT) defines the procedure to be followed if the TMC or Audit record has met the conditions specified by the previous WHEN control statement(s). An ACTION control statement terminates every WHEN control statement.
- The INCLUDE control statement allows the inclusion of control statements from a PDS member. This control statement is optional.

Note: You must specify at least one LIST statement in the PROCESS-SECTION for TMSGRW to select records.

Every ACTION statement (REJECT or LIST) must be preceded by a WHEN statement.

To break a WHEN statement and continue it on a subsequent line, the AND or OR must be the first word on the continued line.

Use of WHEN Statement:

VALID	INVALID
PROCESS-SECTION * TO GET VOLS EXPIRING *TOMORROW WHEN EXPDT NE T-DATE REJECT WHEN VOLSER EQ VOLSER LIST	PROCESS-SECTION * TO GET VOLS EXPIRING *TOMORROW WHEN EXPDT NE T-DATE REJECT LIST
PROCESS-SECTION * TO GET ALL VOLUMES WHEN VOLSER EQ VOLSER LIST	PROCESS-SECTION * TO GET ALL VOLUMES LIST
PROCESS-SECTION * TO GET ACTIVE VOLS WHEN FLAG1 OFF SCR AND FLAG1 OFF DEL LIST	PROCESS-SECTION * TO GET ACTIVE VOLS WHEN FLAG1 OFF SCR AND FLAG1 OFF DEL LIST

PROCESS-SECTION Control Statement

The PROCESS-SECTION control statement identifies the start of the PROCESS-SECTION. This control statement is **required.** Only one PROCESS-SECTION control statement may be submitted per program execution. The format for the PROCESS-SECTION control statement is:

```
PROCESS-SECTION
```

Control Statement Definition

PROCESS-SECTION

Identifies the control statement. It must be followed by a space.

WHEN Control Statement

The WHEN control statement defines the test used to determine if the TMC or Audit record is to be selected for printing. The format for the WHEN control statement is:

```
         [ AND ]          [ AND ]
WHEN le  [ OR  ]  le  ... [ OR  ] le
```

Control Statement Definition

le is either:

- a logical expression in the form:

  ```
  <operand1> <relational expression> <operand2>
  ```

  operand1,operand2

  Must be valid keywords. The length of operand1 would control the length of any compare, and the data type of operand1 would control the type of compare (logical or decimal). Therefore, if operand1 has been defined with a length of 4, only the first four bytes of the record are compared to operand2.

  relational expression

  Must be one of the following:

  EQ - equal

  GE - greater than or equal

  GT - greater than

  LE - less than or equal

  LT - less than

  NE - not equal

- or an expression in the form:

  ```
           { ON  }
  operand1 { OFF } operand2
           { MXD }
  ```

  operand1,operand2

  Are as defined in the preceding logical expression, with the additional constraint that they are 1-byte fields only. Normally, operand1 would be a keyword and operand2 is normally a defined hexadecimal value to test specific bits of operand1. It functions as a mask on operand1.

  ON Specifies all bits on.

  OFF Specifies all bits off.

  MXD Specifies at least one bit on and at least one bit off.

Every WHEN control statement is terminated by an ACTION control statement. You find the valid keywords in the *Programming Guide*. If you use DSNB=YES or DSNB=FAST, the DSNB record is mapped into the volume definitions. The fields are mapped as follows:

DSNB Keyword Mapping using DSNB=FAST

VOLUME	DSNB	Note
	BID	Not mapped, not available for reporting
	ACTIND	Not mapped, not available for reporting
FLAG1	FLAG1	
LSTDSNB	CURRDSNB	
	PREV	Not mapped, not available for reporting
1STDSNB	NEXT	
VOLSER	VOLSER	
	F1STVSN	Not mapped, not available for reporting
NUMDSNB	FILSEQ	
DSN	DSN	
EXPDT	EXPDT	
CDATE	CDATE	
CTIME	CTIME	
CJOB	CJOB	
CSTEP	CSTEP	
LRECL	LRECL	
BLKSIZE	BLKSIZE	
BLKCNT	BLKCNT	
RECFM	RECFM	
BATCHID	BATCHID	
HOOKID	HOOKID	
AUDATE	AUDATE	
AUTIME	AUTIME	
USERID	USERID	
CPUID	CPUID	
AUCODE	AUCODE	

VOLUME	DSNB	Note
AUFLAG1	AUFLAG1	

DSNB Keyword Mapping using DSNB=YES

VOLUME	DSNB	Note
	BID	Not mapped, not available for reporting
	ACTIND	Not mapped, not available for reporting
FLAG1	FLAG1	
LSTDSNB	CURRDSNB	
	PREV	Not mapped, not available for reporting
1STDSNB	NEXT	
	VOLSER	Not mapped, not available for reporting
VOLSER	F1STVSN	
NUMDSNB	FILSEQ	
DSN	DSN	
EXPDT	EXPDT	
CDATE	CDATE	
CTIME	CTIME	
CJOB	CJOB	
CSTEP	CSTEP	
LRECL	LRECL	
BLKSIZE	BLKSIZE	
BLKCNT	BLKCNT	
RECFM	RECFM	
BATCHID	BATCHID	
HOOKID	HOOKID	
AUDATE	AUDATE	
AUTIME	AUTIME	
USERID	USERID	
CPUID	CPUID	
AUCODE	AUCODE	

VOLUME	DSNB	Note
AUFLAG1	AUFLAG1	
OUTCODE		Obtained from the volume and does not have a similar field in the DSNB record.
OUTDATE		Obtained from the volume and does not have a similar field in the DSNB record.
SLOT		Obtained from the volume and does not have a similar field in the DSNB record.

Action Control Statement

The ACTION control statement defines the procedure to be followed if the TMC or Audit record meets the conditions specified by the preceding WHEN control statement. The format for the ACTION control statement is:

```
REJECT
 or
LIST
```

Control Statement Definitions

REJECT

Identifies the control statement. It must be followed by a space. REJECT will prevent the current record from being printed.

LIST

Identifies the control statement. It must be followed by a space. LIST will cause the current record to be printed.

INCLUDE Control Statement

The INCLUDE control statement is discussed earlier in this section.

Control Statements for the REPORT-SECTION

The control statements available for use in the REPORT-SECTION are:

- The REPORT-SECTION control statement identifies the start of the REPORT-SECTION. This control statement is *required*.
- The NEWPAGE control statement will force a top-of-form if the value of a specified field changes. This control statement is optional.

- The PRINT/DUMP/TRANSLATE control statement will cause the specified data contained in the record to be printed under a specified column heading.
- The RDUMP/RLONG control statement causes the entire selected record to be output in a formatted character or hex dump format.
- The INCLUDE control statement allows the inclusion of control statements from a PDS member. This control statement is optional.
- The NOPRINT control statement suppresses report printing. This control statement is optional.

At least one PRINT, DUMP, RDUMP, RLONG, TRANSLATE or NOPRINT control statement must be specified, and they may be interspersed as needed. However, if either RDUMP or RLONG is specified, no other operands are allowed. The control statements should be submitted in the order that the specified fields are to be printed across the page. The total number of characters for the fields requested must not exceed the specified page width. Each field or column is separated by three spaces. The width of each column is determined by the greatest of the following: length of column heading 1, length of column heading 2, or the length of the field in character format.

REPORT-SECTION Control Statement

The REPORT-SECTION control statement identifies the start of the REPORT-SECTION. This control statement is *required*. Only one REPORT-SECTION control statement may be submitted for each program execution. The format for the REPORT-SECTION control statement is:

```
REPORT-SECTION
```

Control Statement Definition

REPORT-SECTION

Identifies the control statement. It must be followed by a space.

NEWPAGE Control Statement

The NEWPAGE control statement forces a top-of-form if the value of the specified field changes. The test for NEWPAGE can be made on the full field or a substring of the field. Only one NEWPAGE control statement is allowed. The format for the NEWPAGE control statement is:

```
NEWPAGE keyword [ON-FIRST nnn]
```

Control Statement Definition

NEWPAGE

Identifies the control statement. It must be followed by a space.

keyword

Identifies the field to be used for the test. When the value changes, it forces a top-of-form. This keyword must be the same keyword specified on the SORT control statement in the CONTROL-SECTION as the major sort field.

ON-FIRST nnn

Indicates that only a partial comparison is to be done on the specified keyword. The value of nnn is numeric, may be from one to three digits, and specifies the actual length of the comparison beginning with the first position of the field. The value of nnn must not exceed the length of the keyword field. For packed fields, nnn=001 will perform a two-digit check. The packed fields are CDATE, CTIME, EXPDT, LDATE, LTIME, OUTDATE, DATECLN and BTHDATE.

PRINT/DUMP/TRANSLATE Control Statement

The PRINT/DUMP/TRANSLATE control statement will cause the specified data contained in the record to be printed under a specified column heading. If a column heading is not specified, the keyword representing that field is used as a column heading. The format for the control statement is:

```
{PRINT    }
{DUMP     }  keyword ['header1[,header2]'] [AS 'date format']
{TRANSLATE}
```

Control Statement Definition

PRINT or DUMP or TRANSLATE

Identifies the control statement. It must be followed by a space. If DUMP is specified, the data contained in the keyword is printed in hexadecimal format.

keyword

Indicates the TMC or Audit field to be printed. The length of a field to be printed can be specified immediately after the field. For example, to print a data set name of 10 characters, enter **PRINT DSN(10)**. To translate hexadecimal values for RECFM, LABEL, DENSITY and TRTCH, specify TRANSLATE instead of PRINT, as in TRANSLATE RECFM 'RECORD,FORMAT'. Valid keywords are the TMC keywords listed in the *Programming Guide* and the *Administration Guide* and the additional TMSGRW keywords listed earlier.

header1,header2

Specifies the column heading for the specified keyword. The column heading must be enclosed within quotes with a comma separating header1 and header2. Two lines are available at output for headings: header2 is printed on the line below header1, and the heading will consist of two lines. If a column heading is not specified, the keyword name is used as header1. For example, if the control statement is PRINT EXPDT, the column heading is EXPDT.

AS 'date format'

Specifies the date format to be printed on reports. This operand is only valid if the keyword being printed is a date keyword. The date format must be specified within delimiters. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21). If CA 1 is active or batch active, this value will override the preferred date format specified in *hlq*.CTAPOPTN or the date format specified on the EXEC PARM; if CA 1 was never initialized, this value will override the default date format 'yyyy/ddd'. This operand is optional.

RDUMP/RLONG Control Statement

RDUMP or RLONG causes the entire selected record to be formatted in a character or hex dump format. The format is identical to that of TMSBINQ. The format of the RDUMP/RLONG statement is:

```
{RDUMP}
{RLONG}
```

Control Statement Definition

RDUMP or RLONG

RDUMP produces a hexadecimal dump that is equivalent to the DUMP format option available with TMSBINQ. RLONG provides a formatted dump of all fields and is equivalent to the LONG format option available with TMSBINQ. Either RDUMP or RLONG can be specified but not both. Neither RDUMP nor RLONG can be included if PRINT, NOPRINT, TRANSLATE, NEWPAGE or DUMP are coded. No other operands are allowed with the RDUMP or RLONG statement.

INCLUDE Control Statement

The INCLUDE control statement is discussed earlier in this section.

NOPRINT Control Statement

The NOPRINT control statement suppresses any requested report printing. The format for the NOPRINT control statement is:

```
NOPRINT
```

Control Statement Definition

NOPRINT

Identifies the control statement. It must be followed by a space, and it must be the last statement in the REPORT-SECTION.

Special Considerations

The total number of DEFINE statements in the CONTROL-SECTION, WHEN/AND/OR statements in the PROCESS-SECTION, and PRINT/DUMP/TRANSLATE statements in the REPORT-SECTION is limited by the block size of the Partitioned Data Set (PDS) member when an INCLUDE statement is used. You can use the following formula to determine this limit:

```
(A*40) + (B*16) + (C*8) + (D*2) < or equal to 72,800
```

where:

- **A** is the number of DEFINE statements
- **B** is the number of instructions
- **C** is the number of print statements
- **D** is the block size of the PDS member used by the INCLUDE statement

Note: Use the block size of the largest PDS member when there is more than one PDS.

Example:

If a PDS member with a block size of 32720 is used by the INCLUDE statement, only 184 DEFINE statements would be allowed.

However, if a PDS member with a block size of 9040 is used, a maximum of 1368 DEFINE statements would be allowed.

TMSGRW Control Statement Examples

Example 1 - Active Data Sets for Jobs XP700000 through XP799999

The following example shows the TMSGRW control statements that will produce a report of all nonscratch data sets created by jobs XP700000 through XP799999. The output is sorted by ascending creating job name, descending data set name, and ascending volume sequence.

The page layout is 40 lines by 104 columns, and secondary files are reported. A page break is generated if the first eight characters of the job name change. The NEWPAGE keyword and the keyword used to specify the major SORT field are the same.

The fields printed are the creating job name, data set name, volume/file sequence, expiration date, and a hex dump of the out-of-area code. Dates are printed in the format 'DDMMMYYYY'. The ALTER statement indicates RUN=SCAN to provide a syntax check of the control statements without report generation. Once the RUN=SCAN subparameter is removed, the report is produced.

According to the delivery instructions, the report is sent to Administration on the second floor.

Example

```
CONTROL-SECTION
*
SORT CJOB D-DSN A-VOLSEQ
TITLE 'ACTIVE DATA SETS FOR JOBS XP700000 THROUGH XP799999'
FOOTER 'SEND REPORT TO ADMINISTRATION, 2ND FLOOR'
ALTER PRINTIMAGE=40X104 DSNB=YES
ALTER INPUT=TMC RUN=SCAN
DEFINE JOB1 CH 'XP700000'
DEFINE JOB2 CH 'XP799999'
*
PROCESS-SECTION
*
WHEN JOB1 GT CJOB
 REJECT
WHEN JOB2 GE CJOB
 LIST
*
REPORT-SECTION
*
NEWPAGE CJOB ON-FIRST 8
PRINT CJOB    'CREATING,JOBNAME'
PRINT DSN     'DATA SET NAME'
PRINT VOLSEQ  'VOLUME,SEQUENCE'
PRINT NUMDSNB 'FILE,SEQUENCE'
PRINT EXPDT   'EXPIRATION,DATE' AS 'DDMMMYYYY'
DUMP  OUTCODE 'HEX,OUTCODE'
```

TMS Report 49 - Instruction Report

The Instruction Report shows the control statements used. It is always printed as the first page of TMSGRW output. Any errors in the control statements are flagged on the Instruction Report, and a list of all TMSGRW errors is produced. TMSGRW error codes and messages are documented later in this section. Once RUN=SCAN is removed from the ALTER control statement, the report is generated as shown in the following report excerpts.

```
TUESDAY, SEPTEMBER 26, 2005.268                  Y O U R   C O M P A N Y   N A M E                             ROxxxxx PAGE=00001

TMS REPORT-49          INSTRUCTION REPORT

  CONTROL-SECTION
  *
  SORT CJOB D-DSN A-VOLSEQ
  TITLE 'ACTIVE DATA SETS FOR JOBS XP700000 THROUGH XP799999'
  FOOTER 'SEND REPORT TO ADMINISTRATION, 2ND FLOOR'
  ALTER PRINTIMAGE=40X104 DSNB=YES
  ALTER INPUT=TMC
  DEFINE JOB1 CH 'XP700000'
  DEFINE JOB2 CH 'XP799999'
  *
  PROCESS-SECTION
  *
  WHEN JOB1 GT CJOB
   REJECT
  WHEN JOB2 GE CJOB
   LIST
  *
  REPORT-SECTION
  *
  NEWPAGE CJOB ON-FIRST 8
  PRINT CJOB    'CREATING,JOBNAME'
  PRINT DSN     'DATA SET NAME'
  PRINT VOLSEQ  'VOLUME,SEQUENCE'
  PRINT NUMDSNB 'FILE,SEQUENCE'
  PRINT EXPDT   'EXPIRATION,DATE' AS 'DDMMMYYYY'
  DUMP  OUTCODE 'HEX,OUTCODE'
```

TMS Report 49 - Active Data Sets for Jobs XP700000 Through XP799999

```
TUESDAY, SEPTEMBER 20, 2005.268                  Y O U R   C O M P A N Y   N A M E                                 PAGE=00002

TMS REPORT-49                          ACTIVE DATA SETS FOR JOBS XP700000 THROUGH XP799999

  CREATING   DATA SET NAME                               VOLUME     FILE       EXPDT        HEX
  JOBNAME                                                SEQUENCE   SEQUENCE                OUTCODE

  XP720000   ISS3.ANALYZER.CNTL                          00001      00001      PERMANENT    00000000
```

```
TUESDAY, SEPTEMBER 20, 2005.268                 Y O U R   C O M P A N Y   N A M E                        PAGE=00003

 TMS REPORT-49                         ACTIVE DATA SETS FOR JOBS XP700000 THROUGH XP799999




 0000000  RECORDS READ BY INCLUDE(S)
 0000021  RECORDS READ BY SYSIN
 0002088  RECORDS READ INTO SORT
 0000001  RECORDS SELECTED FOR OUTPUT
```

```
TUESDAY, SEPTEMBER 20, 2005.268                 Y O U R   C O M P A N Y   N A M E                        PAGE=00004

 TMS REPORT-49                         ACTIVE DATA SETS FOR JOBS XP700000 THROUGH XP799999


                                              SEND REPORT TO ADMINISTRATION, 2ND FLOOR
```

Example 2 - Tapes Expiring Over the Thanksgiving Holiday

The following control statements will provide a list of volumes that will expire during the four day Thanksgiving holiday weekend. The output is sorted by volume serial, data set name, expiration date, and volume sequence number.

The report will provide the following information on each record selected for output: expiration date, volume serial number, volume sequence number and data set name. Dates are printed in the format YYYY/MM/DD.

```
CONTROL-SECTION
*
SORT VOLSER DSN EXPDT VOLSEQ
TITLE 'TAPES EXPIRING OVER THE THANKSGIVING HOLIDAY'
DEFINE LOWDATE DT 'YYYY/MM/DD'
DEFINE HIDATE DT 'YYYY/MM/DD'
*
PROCESS-SECTION
*
WHEN EXPDT GE LOWDATE
  AND EXPDT LE HIDATE
LIST
*
REPORT-SECTION
*
PRINT EXPDT  'EXPIRATION,DATE' AS 'YYYY/MM/DD'
PRINT VOLSER 'VOLUME,SERIAL'
PRINT VOLSEQ 'VOLUME,SEQUENCE'
PRINT DSN    'DATA SET NAME'
```

TMS Report 49 - Tapes Expiring Over the Thanksgiving Holiday

```
 TUESDAY, SEPTEMBER 20, 2005.268                 Y O U R   C O M P A N Y   N A M E                            ROxxxxx PAGE=00001

 TMS REPORT-49          INSTRUCTION REPORT

   CONTROL-SECTION
   *
   SORT VOLSER DSN EXPDT VOLSEQ
   TITLE 'TAPES EXPIRING OVER THE THANKSGIVING HOLIDAY'
   DEFINE LOWDATE DT 'YYYY/MM/DD'
   DEFINE HIDATE DT 'YYYY/MM/DD'
   *
   PROCESS-SECTION
   *
   WHEN EXPDT GE LOWDATE
    AND EXPDT LE HIDATE
   LIST
   *
   REPORT-SECTION
   *
   PRINT EXPDT  'EXPIRATION,DATE' AS 'YYYY/MM/DD'
   PRINT VOLSER 'VOLUME,SERIAL'
   PRINT VOLSEQ 'VOLUME,SEQUENCE'
   PRINT DSN    'DATA SET NAME'
```

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                               PAGE=00002

 TMS REPORT-49                              TAPES EXPIRING OVER THE THANKSGIVING HOLIDAY

   EXPIRATION   VOLUME   VOLUME     DATA SET NAME
   DATE         SERIAL   SEQUENCE

   YYYY/MM/DD   055265   00003      CS.BKUPTSO.UDALL41.M80TS1
   YYYY/MM/DD   055489   00004      CS.BKUPTSO.UDALL41.M80TS1
   YYYY/MM/DD   056141   00005      CS.BKUPTSO.UDALL41.M80TS1
   YYYY/MM/DD   056802   00002      CS.BKUPTSO.UDALL41.M80TS1
   YYYY/MM/DD   057207   00001      CS.BKUPTSO.UDCAL41.M80TS1
```

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                               PAGE=00003

 TMS REPORT-49                              TAPES EXPIRING OVER THE THANKSGIVING HOLIDAY



 0000000  RECORDS READ BY INCLUDE(S)

 0000014  RECORDS READ BY SYSIN

 0001716  RECORDS READ INTO SORT

 0000005  RECORDS SELECTED FOR OUTPUT
```

Example 3 - Tapes with Less Than 100 Blocks of Data

The following control statements will provide a report of all nonscratch single file volumes that are using less than 100 blocks of data on a volume. This may be helpful in determining whether data sets should be moved to DASD to improve processing throughput.

The page layout is 55 lines by 86 columns, and a page break is generated if the first four characters of the data set name change. The NEWPAGE keyword and the keyword used to specify the major SORT field are the same. The report will provide the following information on each record selected for output: data set name, record length, block size and data set block count.

According to the delivery instructions, the report is sent to the runtime improvement group.

```
CONTROL-SECTION
*
SORT DSN
TITLE 'TAPES WITH LESS THAN 100 BLOCKS OF DATA'
FOOTER 'DELIVER TO RUN TIME IMPROVEMENT GROUP'
ALTER PRINTIMAGE=55X086
DEFINE BLOCKS HX '00000064'
DEFINE SCRBIT HX '04'
DEFINE FIRST  CH '000000'
DEFINE NODSNB BI '0000'
*
PROCESS-SECTION
*
WHEN FLAG1  OFF SCRBIT
 AND 1STVOL LT  FIRST
 AND NUMDSNB EQ  NODSNB
 AND BLKCNT LE  BLOCKS
LIST
*
REPORT-SECTION
*
NEWPAGE DSN ON-FIRST 004
PRINT DSN     'DATA SET NAME'
PRINT LRECL   'RECORD,LENGTH'
PRINT BLKSIZE 'BLOCK,SIZE'
PRINT BLKCNT  'BLOCK,COUNT'
```

TMS Report 49 - Tapes with Less than 100 Blocks of Data

```
TUESDAY, SEPTEMBER 27, 2005.268                  Y O U R   C O M P A N Y   N A M E                                    ROxxxxx PAGE=00001

TMS REPORT-49          INSTRUCTION REPORT

  CONTROL-SECTION
  *
  SORT DSN
  TITLE 'TAPES WITH LESS THAN 100 BLOCKS OF DATA'
  FOOTER 'DELIVER TO RUN TIME IMPROVEMENT GROUP'
  ALTER PRINTIMAGE=55X086
  DEFINE BLOCKS HX '00000064'
  DEFINE SCRBIT HX '04'
  DEFINE FIRST  CH '000000'
  DEFINE NODSNB BI '0000'
  *
  PROCESS-SECTION
  *
  WHEN FLAG1  OFF SCRBIT
   AND 1STVOL LT  FIRST
   AND NODSNB EQ  NUMDSNB
   AND BLKCNT LE  BLOCKS
  LIST
  *
  REPORT-SECTION
  *
  NEWPAGE DSN ON-FIRST 004
  PRINT DSN     'DATA SET NAME'
  PRINT LRECL   'RECORD,LENGTH'
  PRINT BLKSIZE 'BLOCK,SIZE'
  PRINT BLKCNT  'BLOCK,COUNT'
```

```
TUESDAY, SEPTEMBER 27, 2005.268                  Y O U R   C O M P A N Y   N A M E                                        PAGE=00002

TMS REPORT-49                                TAPES WITH LESS THAN 100 BLOCKS OF DATA

  DATA SET NAME                                  RECORD       BLOCK        BLOCK
                                                 LENGTH       SIZE         COUNT

   CRDCMG.ALPHA.FDRBKUP.BEOD.G0006V00            0000000000   0000032760   0000000000
```

```
TUESDAY, SEPTEMBER 27, 2005.268                  Y O U R   C O M P A N Y   N A M E                                        PAGE=00003

TMS REPORT-49                                TAPES WITH LESS THAN 100 BLOCKS OF DATA

  DATA SET NAME                                  RECORD       BLOCK        BLOCK
                                                 LENGTH       SIZE         COUNT

   SSDECS.DRAM.MB.SOURCE                         0000000024   0000000024   0000000000
```

```
 TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                                 PAGE=00013

 TMS REPORT-49                               TAPES WITH LESS THAN 100 BLOCKS OF DATA

   DATA SET NAME                                 RECORD       BLOCK        BLOCK
                                                 LENGTH       SIZE         COUNT

   SF0SF01.VMACOUNT.D860818                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D860919                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D860921                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D861104                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D870104                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D870107                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D870111                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D870112                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D870114                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D870130                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D870214                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D870219                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D870225                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D870919                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D871030                      0000000000   0000000000   0000000000
   SF0SF01.VMACOUNT.D871103                      0000000000   0000000000   0000000000
```

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                                 PAGE=00027

 TMS REPORT-49                               TAPES WITH LESS THAN 100 BLOCKS OF DATA



 0000000  RECORDS READ BY INCLUDE(S)

 0000019  RECORDS READ BY SYSIN

 0002288  RECORDS READ INTO SORT

 0000070  RECORDS SELECTED FOR OUTPUT
```

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                                 PAGE=00028

 TMS REPORT-49                               TAPES WITH LESS THAN 100 BLOCKS OF DATA



                                               DELIVER TO RUN TIME IMPROVEMENT GROUP
```

Example 4 - Nonscratch Tapes with No Data

The following control statements will report all nonscratch volumes that are single file and contain no data. This will enable you to identify those programs that are not checking to see if they have data to write before opening the tape for output.

The output is sorted by data set name, creating job name and creation date. The report will provide the following information on each record selected for output: first 10 characters of the data set name, data set block count, volume serial number, creating job name, creating step name, creation date and expiration date. Dates are printed in the format MM/DD/YYYY.

According to the delivery instructions, the report is sent to the production control manager.

```
CONTROL-SECTION
*
SORT DSN CJOB CDATE
TITLE 'NONSCRATCH TAPES WITH NO DATA'
FOOTER 'DELIVER TO PRODUCTION CONTROL MANAGER'
DEFINE SCRBIT HX '04'
DEFINE CLOBIT HX '40'
DEFINE BLOCKS HX '00000000'
DEFINE VSEQ   HX '0001'
DEFINE NODSNB BI '0000'
*
PROCESS-SECTION
*
WHEN FLAG1   OFF SCRBIT
 AND FLAG1   ON  CLOBIT
 AND VSEQ    EQ  VOLSEQ
 AND NUMDSNB EQ  NODSNB
 AND BLOCKS  EQ  BLKCNT
LIST
*
REPORT-SECTION
*
PRINT DSN(10) 'DATA SET NAME'
PRINT BLKCNT  'BLOCK,COUNT'
PRINT VOLSER  'VOLUME,SERIAL'
PRINT CJOB    'CREATING,JOBNAME'
PRINT CSTEP   'CREATING,STEPNAME'
PRINT CDATE   'CREATE,DATE' AS 'MM/DD/YYYY'
PRINT EXPDT   'EXPIRE,DATE' AS 'MM/DD/YYYY'
*
/*
```

TMS Report 49 - Instruction Report

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                          ROxxxxx PAGE=00001

TMS REPORT-49          INSTRUCTION REPORT

  CONTROL-SECTION
  *
  SORT DSN CJOB CDATE
  TITLE 'NONSCRATCH TAPES WITH NO DATA'
  FOOTER 'DELIVER TO PRODUCTION CONTROL MANAGER'
  DEFINE SCRBIT HX '04'
  DEFINE CLOBIT HX '40'
  DEFINE BLOCKS HX '00000000'
  DEFINE VSEQ   HX '0001'
  DEFINE NODSNB HX '0000'
  *
  PROCESS-SECTION
  *
  WHEN FLAG1   OFF SCRBIT
   AND FLAG1   ON  CLOBIT
   AND VSEQ    EQ  VOLSEQ
   AND NUMDSNB EQ  NODSNB
   AND BLOCKS  EQ  BLKCNT
  LIST
  *
  REPORT-SECTION
  *
  PRINT DSN(10) 'DATA SET NAME'
  PRINT BLKCNT  'BLOCK,COUNT'
  PRINT VOLSER  'VOLUME,SERIAL'
  PRINT CJOB    'CREATING,JOBNAME'
  PRINT CSTEP   'CREATING,STEPNAME'
  PRINT CDATE   'CREATE,DATE' AS 'MM/DD/YYYY'
  PRINT EXPDT   'EXPIRE,DATE' AS 'MM/DD/YYYY'
  *
```

TMS Report 49 - Nonscratch Tapes with No Data

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                              PAGE=00002

TMS REPORT-49                                    NONSCRATCH TAPES WITH NO DATA

  DATA SET NAME   BLOCK        VOLUME  CREATING  CREATING   CREATE      EXPIRE
                  COUNT        SERIAL  JOBNAME   STEPNAME   DATE        DATE

  DB2TM02.TM      0000000000   056604  TMSRPT    STEPCOPY   MM/DD/YYYY  MM/DD/YYYY
  SSDECS.UCA      0000000000   056617  DMTGRW    TMSGRW1    MM/DD/YYYY  PERMANENT
  SSDECS.UCA      0000000000   056728  DULCBKPD  BKRL       MM/DD/YYYY  CYCLE/003
```

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                              PAGE=00003

TMS REPORT-49                                    NONSCRATCH TAPES WITH NO DATA



0000000  RECORDS READ BY INCLUDE(S)

0000024  RECORDS READ BY SYSIN

0003432  RECORDS READ INTO SORT

0000003  RECORDS SELECTED FOR OUTPUT
```

```
TUESDAY, SEPTEMBER 27, 2005.268                Y O U R   C O M P A N Y   N A M E                          PAGE=00004

 TMS REPORT-49                                   NONSCRATCH TAPES WITH NO DATA



                                              DELIVER TO PRODUCTION CONTROL MANAGER
```

Example 5 - Nonresident Tape Processing for Yesterday

The following control statements will provide a report of all nonresident tape processing that occurred on the previous day. The Audit data set is used as input. Because the input ddname in the control statement is Audit, no DD statement is needed in the TMSGRW JCL. The active Audit is dynamically allocated and used as input.

The page layout is 55 lines by 90 columns, and the output is sorted by volume serial number, user ID and data set name. The report will provide the following information on each record selected for output: volume serial number, user ID, data set name, CPU ID and exception code.

According to the delivery instructions, the report is sent to the security administrator.

```
CONTROL-SECTION
*
SORT VOLSER USERID DSN
TITLE 'NONRESIDENT TAPE PROCESSING FOR YESTERDAY'
FOOTER 'DELIVER TO SECURITY ADMINISTRATOR'
ALTER INPUT=AUDIT PRINTIMAGE=55X090
DEFINE FTCODE  BI '03'
DEFINE FTEXCP1 CH '003'
DEFINE FTEXCP2 CH '007'
*
PROCESS-SECTION
*
WHEN FTCODE EQ AUCODE
 AND Y-DATE EQ LDATE
 AND AUBLKCO GE FTEXCP1
 AND AUBLKCO LE FTEXCP2
LIST
*
REPORT-SECTION
*
PRINT VOLSER  'VOLUME,SERIAL'
PRINT USERID  'USER,ID'
PRINT DSN     'DATA SET NAME'
PRINT CPUID   'CPU,ID'
PRINT AUBLKCO 'EXCEPTION,CODE'
```

TMS Report 49 - Nonresident Tape Processing for Yesterday

```
FRIDAY, JULY 22, 2005.208                        Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-49          INSTRUCTION REPORT

  CONTROL-SECTION
  *
  SORT VOLSER USERID DSN
  TITLE 'NONRESIDENT TAPE PROCESSING FOR YESTERDAY'
  FOOTER 'DELIVER TO SECURITY ADMINISTRATOR'
  ALTER INPUT=AUDIT PRINTIMAGE=55X090
  DEFINE FTCODE  BI '03'
  DEFINE FTEXCP1 CH '003'
  DEFINE FTEXCP2 CH '007'
  *
  PROCESS-SECTION
  *
  WHEN FTCODE EQ AUCODE
   AND Y-DATE EQ LDATE
   AND AUBLKCO GE FTEXCP1
   AND AUBLKCO LE FTEXCP2
  LIST
  *
  REPORT-SECTION
  *
  PRINT VOLSER  'VOLUME,SERIAL'
  PRINT USERID  'USER,ID'
  PRINT DSN     'DATA SET NAME'
  PRINT CPUID   'CPU,ID'
  PRINT AUBLKCO 'EXCEPTION,CODE'
```

```
FRIDAY, JULY 22, 2005.208                        Y O U R   C O M P A N Y   N A M E                                   PAGE=00002

TMS REPORT-49                                 NONRESIDENT TAPE PROCESSING FOR YESTERDAY

  VOLUME   USER       DATA SET NAME                                  CPU    EXCEPTION
  SERIAL   ID                                                        ID     CODE

  AAAAAA   CAIUSER    TEST.DATA                                      XE90   004
```

```
FRIDAY, JULY 22, 2005.208                        Y O U R   C O M P A N Y   N A M E                                   PAGE=00003

TMS REPORT-49                                 NONRESIDENT TAPE PROCESSING FOR YESTERDAY




0000000  RECORDS READ BY INCLUDE(S)

0000020  RECORDS READ BY SYSIN

0011305  RECORDS READ INTO SORT

0000001  RECORDS SELECTED FOR OUTPUT
```

```
FRIDAY, JULY 22, 2005.208                        Y O U R   C O M P A N Y   N A M E                                   PAGE=00004

TMS REPORT-49                                 NONRESIDENT TAPE PROCESSING FOR YESTERDAY




                                                 DELIVER TO SECURITY ADMINISTRATOR
```

Example 6 - Invalid Use of Permanent Retention

For this example, assume that your data center does not allow permanent retention unless the data set names begin with 'ASM2.ARCHIVE', 'BACKUP.IRS', or BACKUP.FULLVOL. The control statements below will identify the invalid use of permanent retention. The report will list any volumes using permanent retention which do not store data set names beginning with 'ASM2.ARCHIVE', 'BACKUP.IRS' or 'BACKUP.FULLVOL'.

The page layout is 60 lines by 85 columns, and the output is sorted by data set name and user ID. The report will provide the following information on each record selected for output: data set name, volume serial number, user ID and expiration date. Dates are printed in the format MMMDD,YYYY. According to the delivery instructions, the report is sent to the manager of storage resources.

```
CONTROL-SECTION
*
SORT DSN USERID
TITLE 'INVALID USE OF PERMANENT RETENTION'
FOOTER 'DELIVER TO MANAGER OF STORAGE RESOURCES'
ALTER PRINTIMAGE=60X085
DEFINE DSN1 CH   'ASM2.ARCHIVE'
DEFINE DSN2 CH   'BACKUP.IRS'
DEFINE DSN3 CH   'BACKUP.FULLVOL'
DEFINE PERM1 DT  'PERMANENT'
DEFINE PERMLO DT 'STATS/001'
DEFINE PERMHI DT 'STATS/999'
*
PROCESS-SECTION
*
WHEN DSN1 EQ DSN
  OR DSN2 EQ DSN
  OR DSN3 EQ DSN
REJECT
*
WHEN EXPDT EQ PERM1
  OR EXPDT GE PERMLO
 AND EXPDT LE PERMHI
LIST
*
REPORT-SECTION
*
PRINT DSN    'DATA SET NAME'
PRINT VOLSER 'VOLUME,SERIAL'
PRINT USERID 'USER,ID'
PRINT EXPDT  'EXPIRE,DATE' AS 'MMMDD,YYYY'
```

TMS Report 49 - Invalid Use of Permanent Retention

```
TUESDAY, SEPTEMBER 27, 2005.268                  Y O U R   C O M P A N Y   N A M E                          ROxxxxx PAGE=00001

TMS REPORT-49          INSTRUCTION REPORT

  CONTROL-SECTION
  *
  SORT DSN USERID
  TITLE 'INVALID USE OF PERMANENT RETENTION''
  FOOTER 'DELIVER TO MANAGER OF STORAGE RESOURCES'
  ALTER PRINTIMAGE=60X085
  DEFINE DSN1 CH   'ASM2.ARCHIVE'
  DEFINE DSN2 CH   'BACKUP.IRS'
  DEFINE DSN3 CH   'BACKUP.FULLVOL'
  DEFINE PERM1 DT  'PERMANENT'
  DEFINE PERMLO DT 'STATS/001'
  DEFINE PERMHI DT 'STATS/999'
  *
  PROCESS-SECTION
  *
  WHEN DSN1 EQ DSN
    OR DSN2 EQ DSN
    OR DSN3 EQ DSN
  REJECT
  *
  WHEN EXPDT EQ PERM1
    OR EXPDT GE PERMLO
   AND EXPDT LE PERMHI
  LIST
  *
  REPORT-SECTION
  *
  PRINT DSN     'DATA SET NAME'
  PRINT VOLSER  'VOLUME,SERIAL'
  PRINT USERID  'USER,ID'
  PRINT EXPDT   'EXPIRE,DATE' AS 'MMMDD,YYYY'
```

```
TUESDAY, SEPTEMBER 27, 2005.268              Y O U R   C O M P A N Y   N A M E                            PAGE=00002

TMS REPORT-49                               INVALID USE OF PERMANENT RETENTION

  DATA SET NAME                              VOLUME   USER       EXPIRE
                                             SERIAL   ID         DATE

  BACKUP.M90ROS                              056989   TMJEK01    PERMANENT
  BACKUP.M90ROS                              056627   TMJEK01    PERMANENT
  BACKUP.M90ROS                              056623   TMJEK01    PERMANENT
  C491JW1.PROBELOG.LOGA                      055430   TMJRK01    PERMANENT
  C491JW1.PROBELOG.LOGA                      056660   TMJRK01    PERMANENT
  D412MM1.JEP.SMFLOG                         056695   VANAM02    STATS/997
  D468CD1.UCLBAS.BKUPTAPE.FRZ0213.MACLIB     056732   CROLA01    PERMANENT
  EXITDOS.INSTALL.DOC                        056636   BUPOY02    PERMANENT
  ISS3.ANALYZER.CNTL                         056666   RDPSO01    STATS/996
  RRI.TEST.RP0001                            056668   MRPLO01    PERMANENT
  SMPMCS                                     056677   WOMMA02    PERMANENT
  SMPMCS                                     056672   WOMMA02    PERMANENT
  SMPMCS                                     056704   WOMMA02    PERMANENT
  SMPMCS                                     056691   WOMMA02    PERMANENT
  SMPMCS                                     056690   WOMMA02    PERMANENT
  SMPMCS                                     056680   WOMMA02    PERMANENT
  SMPMCS                                     056644   WOMMA02    PERMANENT
  SMPMCS                                     056639   WOMMA02    PERMANENT
  SMPMCS                                     056734   WOMMA02    PERMANENT
  SMPMCS                                     056779   WOMMA02    PERMANENT
  SMPMCS                                     056777   WOMMA02    PERMANENT
  SMPMCS                                     056773   WOMMA02    PERMANENT
  SMPMCS                                     056764   WOMMA02    PERMANENT
  SMPMCS                                     056744   WOMMA02    PERMANENT
  SMPMCS                                     056739   WOMMA02    PERMANENT
  SMPMCS                                     056605   WOMMA02    PERMANENT
  SMPMCS                                     056615   WOMMA02    PERMANENT
  SMPMCS                                     056609   WOMMA02    PERMANENT
  SMPMCS                                     056607   WOMMA02    PERMANENT
  SMPMCS                                     056812   WOMMA02    PERMANENT
  SMPMCS                                     056786   WOMMA02    PERMANENT
  SMPMCS                                     056782   WOMMA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0001V00             056602   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0002V00             056631   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0013V00             056722   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0021V00             056796   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0022V00             056794   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0023V00             056827   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0027V00             056681   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0028V00             056624   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0032V00             056667   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0043V00             056733   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0047V00             056647   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0051V00             056603   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0052V00             056658   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0078V00             056758   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0080V00             056815   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0087V00             056626   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODBTDB.G0090V00             056793   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODDTDB.G0001V00             056637   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODDTDB.G0004V00             056829   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODDTDB.G0005V00             056619   NAMPA02    PERMANENT
  SSDDEV.RPLUS.PRODDTDB.G0020V00             056718   NAMPA02    PERMANENT
                                  .
                                  .
                                  .
                                  .
                                  .
```

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                                 PAGE=00004

 TMS REPORT-49                                  INVALID USE OF PERMANENT RETENTION



 0000000  RECORDS READ BY INCLUDE(S)

 0000025  RECORDS READ BY SYSIN

 0004496  RECORDS READ INTO SORT

 0000082  RECORDS SELECTED FOR OUTPUT
```

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                                 PAGE=00005

 TMS REPORT-49                                  INVALID USE OF PERMANENT RETENTION



                                              DELIVER TO MANAGER OF STORAGE RESOURCES
```

Example 7 - Tapes Out-of-Area to Non-VMS Locations

For this example, assume that the CA 1 Vault Management System (VMS) uses the locations DRA1, SAFE, and PERM, and the VMS pseudo-vaults LIBR and LIB, for off-site storage. The control statements below will produce a report that lists the volumes marked out-of-area to non-VMS locations (locations not owned by the Vault Management System).

You can use this report to inventory the tape library. Tapes that have been returned can be checked in and expired, if desired, to allow them to be reused. Tapes that have been out-of-area for quite some time and have not been returned may need to be replaced if it seems unlikely that they will be returned.

The report will provide the following information on each record selected for output: volume serial number, location ID of the off-site volume, date on which the volume moved off-site, first 10 characters of the data set name and the expiration date. Dates are printed in the format 'DDMMM YYYY'. According to the delivery instructions, the report is delivered to the tape library administrator.

```
CONTROL-SECTION
*
TITLE 'TAPES OUT-OF-AREA TO NON-VMS LOCATIONS'
FOOTER 'DELIVER TO TAPE LIBRARY ADMINISTRATOR'
DEFINE OUT1 CH  ' LIB'
DEFINE OUT2 CH  'LIBR'
DEFINE OUT3 CH  'DRA1'
DEFINE OUT4 CH  'SAFE'
DEFINE OUT5 CH  'PERM'
DEFINE NOOUT HX '00'
*
PROCESS-SECTION
*
WHEN OUT1 NE OUTCODE
 AND OUT2 NE OUTCODE
 AND OUT3 NE OUTCODE
 AND OUT4 NE OUTCODE
 AND OUT5 NE OUTCODE
 AND NOOUT NE OUTCODE
LIST
*
REPORT-SECTION
*
PRINT VOLSER  'VOLUME,SERIAL'
PRINT OUTCODE 'OUT,CODE'
PRINT OUTDATE 'OUT,DATE' AS 'DDMMM YYYY'
PRINT DSN(10) 'DATA SET NAME'
PRINT EXPDT   'EXPIRE,DATE' AS 'DDMMM YYYY'
*
```

TMS Report 49 - Tapes Out-of-Area to Non-VMS Locations

```
TUESDAY, SEPTEMBER 27, 2005.268                  Y O U R   C O M P A N Y   N A M E                             ROxxxxx PAGE=00001

TMS REPORT-49          INSTRUCTION REPORT

  CONTROL-SECTION
  *
  TITLE 'TAPES OUT-OF-AREA TO NON-VMS LOCATIONS'
  FOOTER 'DELIVER TO TAPE LIBRARY ADMINISTRATOR'
  DEFINE OUT1 CH  ' LIB'
  DEFINE OUT2 CH  'LIBR'
  DEFINE OUT3 CH  'DRA1'
  DEFINE OUT4 CH  'SAFE'
  DEFINE OUT5 CH  'PERM'
  DEFINE NOOUT HX '00'
  *
  PROCESS-SECTION
  *
  WHEN OUT1 NE OUTCODE
   AND OUT2 NE OUTCODE
   AND OUT3 NE OUTCODE
   AND OUT4 NE OUTCODE
   AND OUT5 NE OUTCODE
   AND NOOUT NE OUTCODE
  LIST
  *
  REPORT-SECTION
  *
  PRINT VOLSER  'VOLUME,SERIAL'
  PRINT OUTCODE 'OUT,CODE'
  PRINT OUTDATE 'OUT,DATE' AS 'DDMMM YYYY'
  PRINT DSN(10) 'DATA SET NAME'
  PRINT EXPDT   'EXPIRE,DATE' AS 'DDMMM YYYY'
  *
```

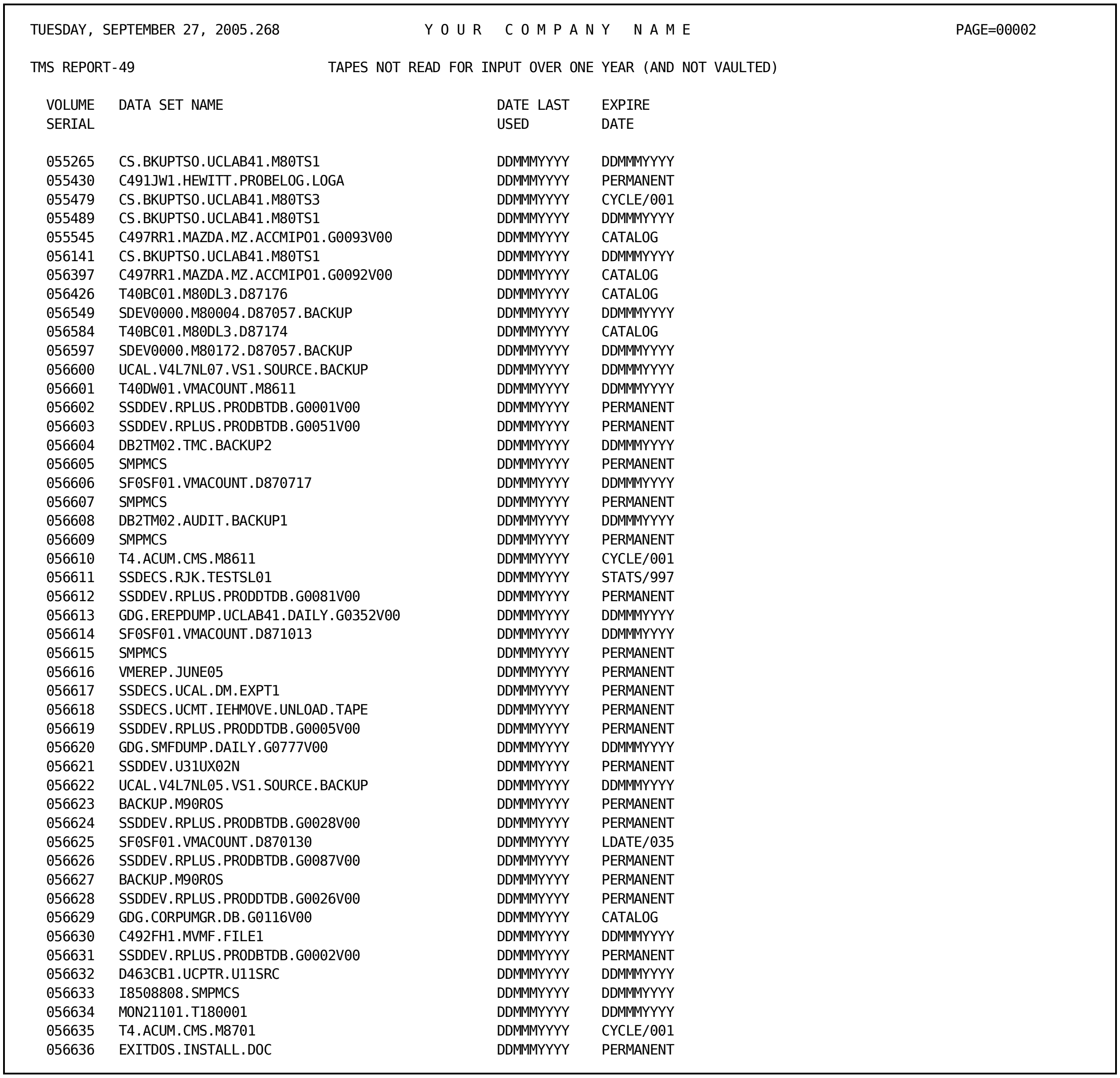
```
TUESDAY, SEPTEMBER 27, 2005.268                Y O U R   C O M P A N Y   N A M E                   PAGE=00002

 TMS REPORT-49                               TAPES OUT-OF-AREA TO NON-VMS LOCATIONS

   VOLUME   OUT    OUT          DATA SET NAME   EXPIRE
   SERIAL   CODE   DATE                         DATE

   056625   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056640   PCC    DDMMM YYYY   RRI.TEST.B      DDMMM YYYY
   056646   WISE   DDMMM YYYY   D468CD1.SY      DDMMM YYYY
   056648   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056668   PCC    DDMMM YYYY   RRI.TEST.R      PERMANENT
   056673   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056676   PCC    DDMMM YYYY   SF0SF01.VM      LDATE/035
   056678   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056685   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056698   HAMI   DDMMM YYYY   CL.PHIL.BI      DDMMM YYYY
   056726   BETL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056736   PCC    DDMMM YYYY   RRI.PROD.R      DDMMM YYYY
   056746   PCC    DDMMM YYYY   CL.PBIS.FS      DDMMM YYYY
   056750   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056756   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056767   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056782   BILL   DDMMM YYYY   SMPMCS          PERMANENT
   056783   PCC    DDMMM YYYY   CL.PBIS.MI      DDMMM YYYY
   056785   PCC    DDMMM YYYY   SF0SF01.VM      LDATE/035
   056788   PCC    DDMMM YYYY   RRI.TEST.B      DDMMM YYYY
   056791   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056792   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056810   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056817   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
   056818   CAGA   DDMMM YYYY   UCPT.R28.G      PERMANENT
   056822   BCTL   DDMMM YYYY   SF0SF01.VM      LDATE/035
```

```
TUESDAY, SEPTEMBER 27, 2005.268                Y O U R   C O M P A N Y   N A M E                   PAGE=00003

 TMS REPORT-49                               TAPES OUT-OF-AREA TO NON-VMS LOCATIONS



 0000000  RECORDS READ BY INCLUDE(S)

 0000025  RECORDS READ BY SYSIN

 0004576  RECORDS READ INTO SORT

 0000026  RECORDS SELECTED FOR OUTPUT
```

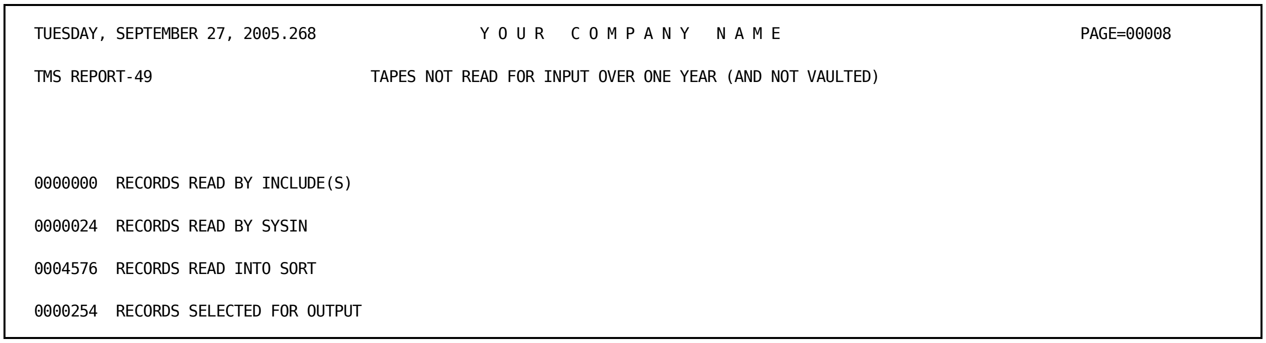
```
TUESDAY, SEPTEMBER 27, 2005.268                Y O U R   C O M P A N Y   N A M E                   PAGE=00004

 TMS REPORT-49                               TAPES OUT-OF-AREA TO NON-VMS LOCATIONS



                                              DELIVER TO TAPE LIBRARY ADMINISTRATOR
```

TMS Report 49 - Tapes Not Read for Input Over One Year (and Not Vaulted)

```
TUESDAY, SEPTEMBER 27, 2005.268                Y O U R   C O M P A N Y   N A M E                           ROxxxxx PAGE=00001

TMS REPORT-49          INSTRUCTION REPORT

  CONTROL-SECTION
  *
  TITLE 'TAPES NOT READ FOR INPUT OVER ONE YEAR (AND NOT VAULTED)'
  FOOTER 'DELIVER TO TAPE LIBRARY ADMINISTRATOR'
  DEFINE OUT1 CH  ' LIB'
  DEFINE OUT2 CH  'LIBR'
  DEFINE OUT3 CH  'DRA1'
  DEFINE OUT4 CH  'SAFE'
  DEFINE OUT5 CH  'PERM'
  DEFINE DATE DT  'DDMMMYYYY'
  DEFINE ZERO DT  'ZEROS'
  *
  PROCESS-SECTION
  *
  WHEN OUT1 NE OUTCODE
   AND OUT2 NE OUTCODE
   AND OUT3 NE OUTCODE
   AND OUT4 NE OUTCODE
   AND OUT5 NE OUTCODE
   AND LDATE LE DATE
   AND LDATE GT ZERO
  LIST
  *
  REPORT-SECTION
  *
  PRINT VOLSER  'VOLUME,SERIAL'
  PRINT DSN     'DATA SET NAME'
  PRINT LDATE   'DATE LAST,USED' AS 'DDMMMYYYY'
  PRINT EXPDT   'EXPIRE,DATE' AS 'DDMMMYYYY'
  *
```



```
TUESDAY, SEPTEMBER 27, 2005.268                  Y O U R   C O M P A N Y   N A M E                                 PAGE=00002

TMS REPORT-49                        TAPES NOT READ FOR INPUT OVER ONE YEAR (AND NOT VAULTED)

  VOLUME   DATA SET NAME                                  DATE LAST    EXPIRE
  SERIAL                                                  USED         DATE

  055265   CS.BKUPTSO.UCLAB41.M80TS1                      DDMMMYYYY    DDMMMYYYY
  055430   C491JW1.HEWITT.PROBELOG.LOGA                   DDMMMYYYY    PERMANENT
  055479   CS.BKUPTSO.UCLAB41.M80TS3                      DDMMMYYYY    CYCLE/001
  055489   CS.BKUPTSO.UCLAB41.M80TS1                      DDMMMYYYY    DDMMMYYYY
  055545   C497RR1.MAZDA.MZ.ACCMIPO1.G0093V00             DDMMMYYYY    CATALOG
  056141   CS.BKUPTSO.UCLAB41.M80TS1                      DDMMMYYYY    DDMMMYYYY
  056397   C497RR1.MAZDA.MZ.ACCMIPO1.G0092V00             DDMMMYYYY    CATALOG
  056426   T40BC01.M80DL3.D87176                          DDMMMYYYY    CATALOG
  056549   SDEV0000.M80004.D87057.BACKUP                  DDMMMYYYY    DDMMMYYYY
  056584   T40BC01.M80DL3.D87174                          DDMMMYYYY    CATALOG
  056597   SDEV0000.M80172.D87057.BACKUP                  DDMMMYYYY    DDMMMYYYY
  056600   UCAL.V4L7NL07.VS1.SOURCE.BACKUP                DDMMMYYYY    DDMMMYYYY
  056601   T40DW01.VMACOUNT.M8611                         DDMMMYYYY    DDMMMYYYY
  056602   SSDDEV.RPLUS.PRODBTDB.G0001V00                 DDMMMYYYY    PERMANENT
  056603   SSDDEV.RPLUS.PRODBTDB.G0051V00                 DDMMMYYYY    PERMANENT
  056604   DB2TM02.TMC.BACKUP2                            DDMMMYYYY    DDMMMYYYY
  056605   SMPMCS                                         DDMMMYYYY    PERMANENT
  056606   SF0SF01.VMACOUNT.D870717                       DDMMMYYYY    DDMMMYYYY
  056607   SMPMCS                                         DDMMMYYYY    PERMANENT
  056608   DB2TM02.AUDIT.BACKUP1                          DDMMMYYYY    DDMMMYYYY
  056609   SMPMCS                                         DDMMMYYYY    PERMANENT
  056610   T4.ACUM.CMS.M8611                              DDMMMYYYY    CYCLE/001
  056611   SSDECS.RJK.TESTSL01                            DDMMMYYYY    STATS/997
  056612   SSDDEV.RPLUS.PRODDTDB.G0081V00                 DDMMMYYYY    PERMANENT
  056613   GDG.EREPDUMP.UCLAB41.DAILY.G0352V00            DDMMMYYYY    DDMMMYYYY
  056614   SF0SF01.VMACOUNT.D871013                       DDMMMYYYY    DDMMMYYYY
  056615   SMPMCS                                         DDMMMYYYY    PERMANENT
  056616   VMEREP.JUNE05                                  DDMMMYYYY    PERMANENT
  056617   SSDECS.UCAL.DM.EXPT1                           DDMMMYYYY    PERMANENT
  056618   SSDECS.UCMT.IEHMOVE.UNLOAD.TAPE                DDMMMYYYY    PERMANENT
  056619   SSDDEV.RPLUS.PRODDTDB.G0005V00                 DDMMMYYYY    PERMANENT
  056620   GDG.SMFDUMP.DAILY.G0777V00                     DDMMMYYYY    DDMMMYYYY
  056621   SSDDEV.U31UX02N                                DDMMMYYYY    PERMANENT
  056622   UCAL.V4L7NL05.VS1.SOURCE.BACKUP                DDMMMYYYY    DDMMMYYYY
  056623   BACKUP.M90ROS                                  DDMMMYYYY    PERMANENT
  056624   SSDDEV.RPLUS.PRODBTDB.G0028V00                 DDMMMYYYY    PERMANENT
  056625   SF0SF01.VMACOUNT.D870130                       DDMMMYYYY    LDATE/035
  056626   SSDDEV.RPLUS.PRODBTDB.G0087V00                 DDMMMYYYY    PERMANENT
  056627   BACKUP.M90ROS                                  DDMMMYYYY    PERMANENT
  056628   SSDDEV.RPLUS.PRODDTDB.G0026V00                 DDMMMYYYY    PERMANENT
  056629   GDG.CORPUMGR.DB.G0116V00                       DDMMMYYYY    CATALOG
  056630   C492FH1.MVMF.FILE1                             DDMMMYYYY    DDMMMYYYY
  056631   SSDDEV.RPLUS.PRODBTDB.G0002V00                 DDMMMYYYY    PERMANENT
  056632   D463CB1.UCPTR.U11SRC                           DDMMMYYYY    DDMMMYYYY
  056633   I8508808.SMPMCS                                DDMMMYYYY    DDMMMYYYY
  056634   MON21101.T180001                               DDMMMYYYY    DDMMMYYYY
  056635   T4.ACUM.CMS.M8701                              DDMMMYYYY    CYCLE/001
  056636   EXITDOS.INSTALL.DOC                            DDMMMYYYY    PERMANENT
```



```
TUESDAY, SEPTEMBER 27, 2005.268                  Y O U R   C O M P A N Y   N A M E                                 PAGE=00008

TMS REPORT-49                        TAPES NOT READ FOR INPUT OVER ONE YEAR (AND NOT VAULTED)



0000000  RECORDS READ BY INCLUDE(S)

0000024  RECORDS READ BY SYSIN

0004576  RECORDS READ INTO SORT

0000254  RECORDS SELECTED FOR OUTPUT
```

```
TUESDAY, SEPTEMBER 27, 2005.268                  Y O U R   C O M P A N Y   N A M E                          PAGE=00009

 TMS REPORT-49                     TAPES NOT READ FOR INPUT OVER ONE YEAR (AND NOT VAULTED)




                                            DELIVER TO TAPE LIBRARY ADMINISTRATOR ErrorsaDuring Label/s That Had I/O Errors During
Label Creation
```

If an I/O error occurs during the system creation of a tape header label, CA 1 moves a special keyword data set name into the volume record. These tapes should be checked to determine if the media is bad, and whether the volume should be cleaned or replaced. The TMSGRW control statements can be used to create a list of tapes with the ** IO ERROR ** data set name.

The report will provide the following information on each record selected for output: volume serial number, unit address last use, date last used and data set name. Dates are printed in the format 'DDMMMYYYY'.

```
CONTROL-SECTION
*
TITLE 'TAPES THAT HAD I/O ERRORS DURING LABEL CREATION'
FOOTER 'DELIVER TO TAPE LIBRARY ADMINISTRATOR'
DEFINE IODSN HX '00405C5C4040C9D6'
*
PROCESS-SECTION
*
WHEN IODSN EQ DSN
LIST
*
REPORT-SECTION
*
PRINT VOLSER 'VOLUME,SERIAL'
PRINT LUNIT 'LAST,UNIT'
PRINT LDATE 'DATE LAST,USED' AS 'DDMMMYYYY'
PRINT DSN 'DATA SET NAME'
```

Example 8 - Tapes Not Read for Input for Over One Year (and Not Vaulted)

This example will provide a list of volumes that have not been read for input for at least a year and are not marked out-of-area to 'LIB', 'LIBR', 'DRA1', 'SAFE', or 'PERM', which are the off-site storage locations owned by the CA 1 Vault Management System (VMS).

The report will provide the following information on each record selected for output: volume serial number, data set name, last used date, and expiration date. Dates are printed in the format DDMMMYYYY. According to the delivery instructions, the report is sent to the tape library administrator.

```
CONTROL-SECTION
*
TITLE 'TAPES NOT READ FOR INPUT OVER ONE YEAR (AND NOT VAULTED)'
FOOTER 'DELIVER TO TAPE LIBRARY ADMINISTRATOR'
DEFINE OUT1 CH  ' LIB'
DEFINE OUT2 CH  'LIBR'
DEFINE OUT3 CH  'DRA1'
DEFINE OUT4 CH  'SAFE'
DEFINE OUT5 CH  'PERM'
DEFINE DATE DT  'DDMMMYYYY'
DEFINE ZERO DT  'ZEROS'
*
PROCESS-SECTION
*
WHEN OUT1 NE OUTCODE
 AND OUT2 NE OUTCODE
 AND OUT3 NE OUTCODE
 AND OUT4 NE OUTCODE
 AND OUT5 NE OUTCODE
 AND LDATE LE DATE
 AND LDATE GT ZERO
LIST
*
REPORT-SECTION
*
PRINT VOLSER 'VOLUME,SERIAL'
PRINT DSN    'DATA SET NAME'
PRINT LDATE  'DATE LAST,USED' AS 'DDMMMYYYY'
PRINT EXPDT  'EXPIRE,DATE' AS 'DDMMMYYYY'
```

TMS Report 49 - Tapes That Had I/O Errors During Label Creation

```
 TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                           ROxxxxx PAGE=00001

 TMS REPORT-49          INSTRUCTION REPORT

   CONTROL-SECTION
   *
   TITLE 'TAPES THAT HAD I/O ERRORS DURING LABEL CREATION'
   FOOTER 'DELIVER TO TAPE LIBRARY ADMINISTRATOR'
   DEFINE IODSN HX '00405C5C4040C9D6'
   *
   PROCESS-SECTION
   *
   WHEN IODSN EQ DSN
   LIST
   *
   REPORT-SECTION
   *
   PRINT VOLSER  'VOLUME,SERIAL'
   PRINT LUNIT   'LAST,UNIT'
   PRINT LDATE   'DATE LAST,USED' AS 'DDMMMYYYY'
   PRINT DSN     'DATA SET NAME'
   *
```

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                              PAGE=00002

 TMS REPORT-49                          TAPES THAT HAD I/O ERRORS DURING LABEL CREATION

   VOLUME   LAST   DATE LAST    DATA SET NAME
   SERIAL   UNIT   USED

   056711   0881   DDMMMYYYY      **  IO ERROR  *
   056725   0883   DDMMMYYYY      **  IO ERROR  *L41.DAILY.G0280V00
```

```
TUESDAY, SEPTEMBER 20, 2005.268                 Y O U R   C O M P A N Y   N A M E                        PAGE=00003

 TMS REPORT-49                           TAPES THAT HAD I/O ERRORS DURING LABEL CREATION



 0000000  RECORDS READ BY INCLUDE(S)

 0000012  RECORDS READ BY SYSIN

 0001144  RECORDS READ INTO SORT

 0000002  RECORDS SELECTED FOR OUTPUT
```

```
TUESDAY, SEPTEMBER 27, 2005.268                 Y O U R   C O M P A N Y   N A M E                        PAGE=00004

 TMS REPORT-49                           TAPES THAT HAD I/O ERRORS DURING LABEL CREATION



                                               DELIVER TO TAPE LIBRARY ADMINISTRATOR
```

TMS Report 49 - Tapes with EXPDT=CATALOG Control that are Active in TMC

```
 THURSDAY JANUARY 22, 2015               Y O U R   C O M P A N Y   N A M E                         ROxxxxx PAGE=00001

TMS REPORT-49          INSTRUCTION REPORT

   CONTROL-SECTION
*
   SORT A-DSN A-VOLSEQ A-CDATE
   TITLE 'TAPES WITH EXPIRATION DATE OF CATALOG'
   DEFINE CATDATE DT 'CATALOG'
*
   PROCESS-SECTION
*
   WHEN EXPDT EQ CATDATE
   LIST
*
   REPORT-SECTION
*
   PRINT VOLSER
   PRINT EXPDT
   PRINT DSN
   PRINT CDATE
```

```
 THURSDAY JANUARY 22, 2015               Y O U R   C O M P A N Y   N A M E                         PAGE=00002

TMS REPORT-49                                 TAPES WITH EXPIRATION DATE OF CATALOG
  VOLSER   EXPDT        DSN                                        CDATE


   760142   CATALOG      MORT002.GDGV.G0072V00                      01-12-2015
   760168   CATALOG      MORT002.GDGV.G0073V00                      01-22-2015
   760169   CATALOG      MORT002.GDGV.G0073V00                      01-22-2015
   760170   CATALOG      MORT002.GDGV.G0073V00                      01-22-2015
   760171   CATALOG      MORT002.GDGV.G0073V00                      01-22-2015
 THURSDAY JANUARY 22, 2015                Y O U R   C O M P A N Y   N A M E                          PAGE=00003
TMS REPORT-49                                 TAPES WITH EXPIRATION DATE OF CATALOG

 0000000  RECORDS READ BY INCLUDE(S)
 00000012  RECORDS READ BY SYSIN
 00002200  RECORDS READ INTO SORT
 00000005  RECORDS SELECTED FOR OUTPUT
```

Example 9 - Tapes with EXPDT=CATALOG Control and are Active in TMC

This example provides a list of tapes with the EXPDT=CATALOG control that are active in the TMC.

```
TMS REPORT-49          INSTRUCTION REPORT
*
   CONTROL-SECTION
*
   SORT A-DSN A-VOLSEQ A-CDATE
   TITLE 'TAPES WITH EXPIRATION DATE OF CATALOG'
   DEFINE CATDATE DT 'CATALOG'
*
   PROCESS-SECTION
*
   WHEN EXPDT EQ CATDATE
   LIST
*
   REPORT-SECTION
*
   PRINT VOLSER
   PRINT EXPDT
   PRINT DSN
   PRINT CDATE
```

TMSGRW Report Field Definitions

Control statement image

Input control statement

RECORDS READ BY INCLUDE(S)

Number of control statements processed through the DD statement(s) that are referenced by INCLUDE control statements

RECORDS READ BY SYSIN

Number of control statements processed through the SYSIN DD

RECORDS READ INTO SORT

Number of records processed by the SORT routine

RECORDS SELECTED BY OUTPUT

Number of records selected for reporting

TMSIDATA Utility—Build a TMC from VSAM or Operating System Catalog Listing

Use the TMSIDATA batch utility program to assist new installations of CA 1 in building the initial TMC when an existing tape management system is not available for conversion.

The TMSIDATA program accepts as input list-only data sets from existing VSAM catalogs. Utilizing the keyed update capability of VSAM, TMSIDATA formats the data and builds correctly chained Volume and DSNB records for subsequent processing by TMSFORMT and TMSAGGR, respectively. You do not have to run TMSMULV to chain multivolume pointers. TMSIDATA restricts input data to IDCAMS and/or IEHLIST catalog listings. No other user data is accepted.

CA 1 can be active, batch active, or inactive to run TMSIDATA. If a TMC is currently active, it must be a nonextended TMC.

Exit Processing

During TMSIDATA processing, the following exits are invoked:

- TMSXITU
- TMSXITE
- EXITPGM

Two exit routines, TMSXITU and TMSXITE, are provided in CA 1 to accommodate volume serial numbers that are not six-digit numerics. As each volume record is processed, TMSXITU is invoked to perform the required alphanumeric to numeric translation. The six-digit numeric volume serial number is used *internally* by CA 1 to access the volume record in the TMC. The nonnumeric volume serial number is still used *externally* to reference the volume.

- If the volume serial number is not numeric after TMSXITU processing, an error message is printed and the record is bypassed.
- If the volume serial number is numeric after TMSXITU processing, the TMSXITE exit is invoked to translate the numeric back to the external volume serial number. TMSIDATA compares this number to the input volume serial number from the catalog listing. This comparison check ensures that your TMSXITU and TMSXITE exits are functioning correctly. If the numbers are different, an error message is printed and the record is bypassed.

Resolve any error messages that are generated during TMSXITU or TMSXITE processing before you continue building the TMC.

Important! TMSXITU, TMSXITE, and the optional output Volume record exit must reside in STEPLIB or must be in a link listed data set.

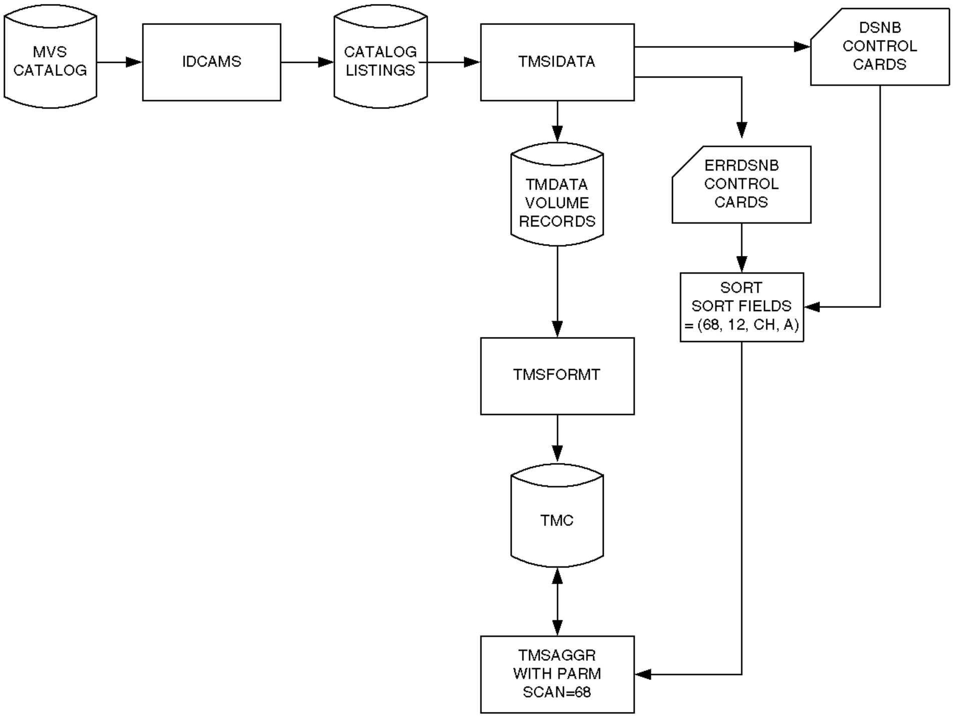
The optional EXITPGM= user exit can be coded to modify the converted TMC data, to fill additional TMC fields or to bypass volumes from the conversion. If specified, this exit is called before each data record is written to the VOLUME or DSNB output data set.

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

Report Description

TMSIDATA generates TMS Report 35 - Build TMC Records From Catalog Listings. This report lists all error and status messages, the number of Volume and DSNB records formatted, the number of IDCAMS lines processed and the number of errors encountered during processing. A sample of this report appears on TMS Report 35 - Build TMC Records From Catalog Listings (see page 327).

The following diagram illustrates Building the TMC Using TMSIDATA:



Job Control Statements

```
//STEP1    EXEC PGM=IDCAMS                               *NOTES
//SYSPRINT DD  DSN=IDCAMS.LIST,
//             DISP=(NEW,CATLG,DELETE),UNIT=SYSDA,
//             SPACE=(CYL,(10,2))
//SYSIN    DD  *
 LISTCAT NONVSAM ALL CAT(ICF.VMVXE90)
 LISTCAT NONVSAM ALL CAT(CATALOG.BRES02)
/*
//STEP3    EXEC PGM=TMSIDATA,
//             [PARM='[DATEFMT=(fmt)][,MAXERR=nnnnn] [,EXITPGM=exitname]
//             [,TMSXITE=exitname][,TMSXITU=exitname]  *NOTES
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)      *NOTES
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK06 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//IDCAMS   DD  DSN=IDCAMS.LIST,DISP=SHR
//IEHLIST  DD  DSN=IEHLIST.LIST,DISP=SHR                 *NOTES
//VOLUME   DD  DSN=CAI.CA1.VOLUME,
//             DISP=(NEW,CATLG,DELETE),UNIT=SYSDA,
//             SPACE=(CYL,(10,2))
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(10,2))
//DSNB     DD  DSN=CAI.CA1.DSNB,
//             DISP=(NEW,CATLG,DELETE),UNIT=SYSDA,
//             SPACE=(CYL,(10,2))
//ERRDSNB  DD  DSN=CAI.CA1.ERRDSNB,
//             DISP=(NEW,CATLG,DELETE),UNIT=SYSDA,
//             SPACE=(CYL,(10,2))
//SYSPRINT DD  SYSOUT=A
//SYSOUT   DD  SYSOUT=A
//SYSIN    DD  *
 (control statements here)
/*
//
```

Notes:

- Before TMSIDATA can run, create catalog listings of the catalogs to be converted. STEP1 shows how to create an IDCAMS catalog listing. Do *not* change the default record format (RECFM) and record length (LRECL) of the IDCAMS list data set. This data set must be RECFM=VBA and LRECL=125. A larger block size (BLKSIZE) can be used to improve performance.
- When using Alpha Numeric exits TMSXITU and TMSXITE or EXITPGM=, specify your assembled exit names as a parameter.
- Sufficient sort work space must be available to accommodate all Volume records to be created. Volume records are 340 bytes.
- If there is no IEHLIST input, specify DUMMY.

Parameter Definitions

DATEFMT=(fmt)

Specifies the default date format to be used for output records. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see the *Utilities and Reports Reference Guide*. This parameter is optional.

MAXERR={*500*|nnnnn}

Specifies the number of errors accepted before the job is terminated with a 1021 abend. The value of nnnnn can have a maximum value of 99999. If this parameter is omitted, up to 500 errors (the default) is accepted before the job abends. This parameter is optional.

XITU=

Identifies the user exit load module name that TMSIDATA uses for alphanumeric volume serial number conversions. The name can be a maximum of eight bytes and must follow MVS naming conventions. This parameter is optional. If CA 1 is active, the active version of XITU will be used instead of what is specified here.

XITE=

Identifies the user exit load module name that TMSIDATA uses for alphanumeric volume serial number conversions. The name can be a maximum of eight bytes and must follow MVS naming conventions. This parameter is optional. If CA 1 is active, the active version of XITU will be used instead of what is specified here.

EXITPGM=

Identifies the user exit load module name that TMSIDATA uses before writing each TMC Volume output record. This exit enables you to modify any field within a Volume record, except the volume serial number itself. This parameter is optional.

JCL Considerations

TMSRPT

Contains error and status messages generated during TMSIDATA processing for TMS Report 35. This ddname is **required**.

IDCAMS

Contains the IDCAMS catalog listing data set. This ddname is **required**.

IEHLIST

Contains the optional IEHLIST catalog listing. If there is no IEHLIST input, DUMMY must be specified. This ddname is **required**.

VOLUME

Contains the formatted Volume records to be input to TMSFORMT. This ddname is **required**.

SYSUT1

Defines a temporary data set to be used for sorting DSNB records. This ddname is **required**.

DSNB

Contains the formatted DSNB records to be input to TMSAGGR. See ERRDSNB. This ddname is **required**.

ERRDSNB

Contains the error DSNB records which, optionally, can be manually corrected and merged with the DSNB records on DD DSNB for input to TMSAGGR. This ddname is **required**.

Note: When you run TMSA GGR, the TMSAGGR execute statement *must* include the 'SCAN=68' parameter.

SYSPRINT

Contains VSAM delete/define messages from IDCAMS. This ddname is **required**.

SYSOUT

Contains messages from your sort program. This ddname is **required**.

SYSIN Control Parameters

The SYSIN control parameters for TMSIDATA are used to dynamically define and allocate two VSAM work data sets, VWORK1 (the ddname for Volume records) and VWORK2 (the ddname for DSNB records). VWORK1 and VWORK2 are used to properly chain Volume and DSNB records.

Note: At the beginning of the TMSIDATA process, IDCAMS DELETE commands are issued for these data sets to prevent the possible duplication of data set names. However, VWORK1 and VWORK2 are left intact at the end of processing. You can manually delete these VSAM data sets or use them for further analysis. The record layouts for VWORK1 and VWORK2 are listed under the control parameter definitions for WKVOL1 and WKVOL2.

Each control parameter specification must begin with a keyword in position 1. A given parameter must be entirely contained on one statement; no continuations are allowed. As many parameter statements as necessary may be coded. The maximum length of data on a SYSIN statement is 79 positions followed by a blank.

The SYSIN control parameters can be coded using either format listed below:

```
 1
 NUMVOL=nnnnnn
 NUMDSN=nnnnnnn
 WKVOL1=xxxxxx
 WKVOL2=xxxxxx
 INDXVOL=xxxxxx
 WKNODE=xxxxxx
[EXPDT={keyword|preferred date|Julian date}]
```

Or

```
 1
 NUMVOL=nnnnnn,NUMDSN=nnnnnnn,WKVOL1=xxxxxx,WKVOL2=xxxxxx,INDXVOL=xxxxxx
 WKNODE=xxxxxx,[EXPDT={keyword|preferred date|Julian date}]
```

Control Parameter Definitions

NUMVOL=nnnnnn

Specifies the anticipated number of Volume records to be processed by TMSIDATA, where nnnnnn is an integer that can have a maximum value of 999999. This number is used to properly size the VSAM work data set (VWORK1) for Volume records. Secondary extents are calculated at 10 percent of the NUMVOL value. Generally, VSAM can extend a keyed sequential data set up to 127 times, depending on the DASD space available. This parameter is **required**.

NUMDSN=nnnnnnn

Specifies the anticipated number of DSNB records to be processed by TMSIDATA, where nnnnnnn is an integer that can have a maximum value of 2500000. This number is used to properly size the VSAM work data set (VWORK2) for DSNB records. Secondary extents are calculated at 10 percent of the NUMDSN value. Generally, VSAM can extend a keyed sequential data set up to 127 times, depending on the DASD space available. This parameter is **required**.

WKVOL1=xxxxxx

Specifies the DASD volume on which to allocate the VSAM work data set (VWORK1) for Volume records, where xxxxxx is a one- to six-character alphanumeric value. This parameter is **required**.

The record layout for VWORK1 is:

```
V1AREA   DS 0CL97
V1KEY    DS 0CL6 VSAM key field
V1VOLSER DS CL6 volume serial number
V1VOLSEQ DS XL2 volume sequence number
V1DSN    DS CL44 data set name
V1DSN17  DS CL17 last 17 characters of data set name
V1CREATE DS PL4 data set creation date
V1EXPIRE DS PL4 expiration date
V1FIRST  DS CL6 first volume pointer
V1PREV   DS CL6 previous volume pointer
V1NEXT   DS CL6 next volume pointer
V1DEN    DS XL1 density
V1TRTCH  DS XL1 recording technique
```

WKVOL2=xxxxxx

Specifies the DASD volume on which to allocate the VSAM work data set (VWORK2) for DSNB records, where xxxxxx is a one- to six-character alphanumeric value. For best performance, the two VSAM work data sets should be on separate volumes. This parameter is **required**. The record layout for VWORK2 is:

```
V2AREA   DS 0CL72
V2KEY    DS 0CL8 VSAM key field
V2LSTVOL DS CL6 last volume that data set resides on
V2FILSEQ DS XL2 file sequence number of data set
V2DSN    DS CL44 data set name
V2MOTHER DS CL6 DSNB mother volume serial
V21STVOL DS CL6 DSNB first volume pointer
V2CREATE DS PL4 creation date
V2EXPIRE DS PL4 expiration date
```

TMSIDATA will *not* delete the two VSAM work data sets when the process finishes. These record layouts are provided so that you can, if desired, report on or analyze the data sets. In the event of a rerun, however, a DELETE is issued at the beginning of the process.

INDXVOL=xxxxxx

Specifies the DASD volume on which to allocate the VSAM indexes for the VWORK1 and VWORK2 VSAM data sets, where xxxxxx is a one- to six-character alphanumeric value. For best performance, this volume should be different from both VWORK1 and VWORK2. This parameter is **required**.

WKNODE=xxxxxx

Specifies a high-level qualifier(s) for naming the VWORK1 and VWORK2 VSAM data sets so that they comply with your site-specific naming conventions. The value of xxxxxx can be a maximum of 17-bytes, and no period is allowed at the end of this parameter. TMSIDATA appends an appropriate suffix to the WKNODE to complete the data set name, as illustrated in the following examples.

```
xxxxxx.CA1.IDATA.DSNCLUST
xxxxxx.CA1.IDATA.VOLCLUST
```

This parameter must follow standard MVS data set naming conventions and is **required**.

EXPDT={keyword|preferred date|Julian date}

Specifies a default expiration date for the Volume records to be created. This value can be a CA 1 keyword, an actual date in your preferred date pattern, or a date in Julian format (yyyy/ddd). If this parameter is omitted, TMSIDATA uses CATALOG as the default expiration date. This parameter is optional.

TMS Report 35 - Build TMC Records From Catalog Listings

```
DATES=07/09/96.190                               C O M P U T E R   A S S O C I A T E S                             ROxxxxx PAGE=00001

TMS REPORT-35     BUILD TMC RECORDS FROM CATALOG LISTINGS


 TMSMSG09  EXEC PARM OPTIONS IN EFFECT:  EXITPGM=        ,  MAXERR=99999,  DATEFMT=MMMDD YYYY


 *** TMSIDATA SYSIN CONTROL PARMS ***

 WKNODE=SYSPROG.TST
 WKVOL1=M80004
 WKVOL2=M80004
 INDXVOL=M80005
 NUMVOL=500
 NUMDSN=600
 EXPDT=CATALOG



 TMSMSG17  VOL-SER, A00000, NOT NUMERIC AFTER TMSXITU XLATION.  VOL=A00000 DSN=TEST.TMSXITU.FILE1
 TMSMSG17  VOL-SER, A00001, NOT NUMERIC AFTER TMSXITU XLATION.  VOL=A00001 DSN=TEST.TMSXITU.FILE2
 TMSMSG17  VOL-SER, A00003, NOT NUMERIC AFTER TMSXITU XLATION.  VOL=A00003 DSN=TEST.TMSXITU.FILE3
 TMSMSG17  VOL-SER, A0CC04, NOT NUMERIC AFTER TMSXITU XLATION.  VOL=A0CC04 DSN=TEST.TMSXITU.FILE4
 TMSMSG02  NOW PROCESSING CATALOG LISTING FOR ---> ICF.VMVXE90
 TMSMSG18  DUPLICATE DSNAME, RECORD BYPASSED. VOL=980007 VSEQ=001 DSN=SSDDEV.CA1.CL050B6.DS01
 TMSMSG12  VOL REC FOR 980007 VSEQ=001 DSN=SSDDEV.CA1.CL050B6.DS01                      REPL BY DSN=SSDDEV.CA1.CL050B6.DS06
 TMSMSG14  VOL IN MULTI-FILE STRING, BUT MOTHER VOL NOT FOUND. VOL=772371 FSEQ=002 DSN=SSDDEV.CA1.CL050B6.AA01
 TMSMSG12  VOL REC FOR 980008 VSEQ=001 DSN=SSDDEV.CA1.CL050B6.DS02                      REPL BY DSN=SSDDEV.CA1.CL050B6.DS02
 TMSMSG12  VOL REC FOR 980009 VSEQ=001 DSN=SSDDEV.CA1.CL050B6.DS03                      REPL BY DSN=SSDDEV.CA1.CL050B6.DS04
 TMSMSG05  VOLSER 980009 PART OF MULTI-FILE STRING, BUT -1 DSNB REC NOT FOUND. FSEQ=007.  DSN=SSDDEV.CA1.CL050B6.DS07



 TMSMSG11A  TOTAL VOLUME RECORDS FORMATTED=00000000018
 TMSMSG11B  TOTAL DSNB RECORDS FORMATTED=00000000006
 TMSMSG11C  TOTAL IDCAMS LINES READ=00000000221
 TMSMSG11D  TOTAL IEHLIST LINES READ=00000000000
 TMSMSG11E  TOTAL ERRORS=00000000007



 TMSIDATA PROCESS SUCCESSFULLY COMPLETED


         END OF REPORT 35  -  TOTAL ITEMS LISTED=        30
```

TMSIDATA Report Field Definitions

EXEC PARM OPTIONS IN EFFECT:

Parameters for TMSIDATA from EXEC JCL statement

***** TMSIDATA SYSIN CONTROL PARMS *****

Input control statements

Message

Processing message

TOTAL VOLUME RECORDS FORMATTED=

Number of Volume records written to output

TOTAL DSNB RECORDS FORMATTED=

Number of DSNB records written to output

TOTAL IDCAMS LINES READ=

Number of IDCAMS lines processed for input

TOTAL IEHLIST LINES READ=

Number of IEHLIST lines processed for input

TOTAL ERRORS=

Number of processing errors

TMSINIT Utility—Activate, Deactivate CA 1

Use the TMSINIT utility to activate, batch activate or deactivate CA 1 on the operating system. TMSINIT also initiates the CA 1 health checks.

CAIRIM initialization activities for CA 1 must have completed successfully for TMSINIT to execute properly. CA 1 can be present in your operating system in one of the following four states:

- In an active state, all CA 1 batch and online facilities are fully functional. The CA 1 OSI is installed and tape activity is tracked.
- In a batch active state, all CA 1 batch and online facilities are fully functional. No realtime tape activity is tracked. The OSI is restored if it is currently installed.
- In an inactive state, some CA 1 batch facilities but none of the CA 1 online facilities are functional. No realtime tape activity is tracked. The OSI is restored if it is currently installed.
- In an *uninitialized* state, a few CA 1 batch facilities but none of the CA 1 online facilities are functional. Any jobs requesting tape I/O are abended (S8xx). Most CA 1 utilities will abend with U0011. The uninitialized state occurs when the operating system has been IPLed and TMSINIT has not been executed to place CA 1 into an active, batch active or inactive state. If the subsystem and TMSINIT have not been started, each tape open will receive IECTMS0.

  Note: For a description of this message and correction action, see the *Messages Reference Guide.*

Except for the first execution after an IPL to activate CA 1, all calls to TMSINIT will result in an additional call to the security system. If the first execution of TMSINIT after an IPL is to deactivate or batch activate CA 1, an additional call to the security system is made. If TMSINIT is run as a batch job, the security authorization of the job is used to determine access. If TMSINIT is run as a started task, a WTOR is issued to obtain the operator's user ID and password. This information is used to determine if the operator has the correct authority to reinitialize, batch-activate or deactivate CA 1. Set the SECWTO option in the TMOOPTxx member of CTAPOPTN to NO if you do not want to validate that the operator executing TMSINIT as a started task has the correct level of authority. This will allow TMSINIT to be executed as a started task by anyone with access to a console.

TMSINIT is used to activate CA 1, and is typically included as part of the system IPL procedures. Deactivation and batch activation of CA 1 can also be performed with this utility. TMSINIT, as a started task, does not stay active in the system once CA 1 is initialized. The system console is issued WTORs for operator response by the SVCs called by TMSINIT, such as:

```
IEFTMS0 REPLY 'U' FOR CA 1. rnn.n INITIALIZATION OR HELP
```

Where *nn.n* is the CA 1 release number.

Important! The IEFTMS0 message should not be responded to by an automated console manager, such as CA-Opera, since there are variable responses to this WTOR. In certain types of critical situations, several executions of TMSINIT may be needed with a response other than U. If an automated console manager is used, then PARM=NODATE should be specified for normal TMS initialization. *Do not* call the started task for TMSINIT execution TMS, as this is the CA 1 subsystem name.

Note: If you have other vendor products which modify the operating system data management modules and these modified versions MLPA'd, you must include the CAG8LIB2 DD in the CAIRIM procedure and the TMSINIT procedure. Any MLPA data set must be specified in the CAG8LIB2 DD which contains these modifications. If you have a data set name for the contents of the LPA other than SYS1.LPALIB, you must put that data set name in the CAG8LIB1 DD and include the DD in the CAIRIM procedure and the TMSINIT procedure. Ensure that both the CAIRIM procedure (CAS9) and the TMSINIT procedure have security access (READ) to the libraries containing the operating system's data management module. If all of the open/close routines are not copied to the modified library, then you must have SYS1.LPALIB in the concatenation with CAG8LIB1.

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 if it is available. The CAHCHECK address space comes up very early in the IPL and normally remains up for the entire IPL. Thus, CA 1 health checks that are hosted by the CAHCHECK address space monitor CA 1 for the life of the IPL.

If the CAHCHECK address space is not available, TMSINIT will not initiate the CA 1 health checks. In order to initiate and host the CA 1 health checks, the CTS address space must be started. If CTS is started in an environment where the CAHCHECK address space is not available, CTS will automatically start the CA 1 health checks.

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

CA 1 must be active before other software packages that use tape or look for the CA 1 mount messages. These include but are not limited to OAM and HSM.

Defining and Changing Installation Options

When TMSINIT is run, *hlq*.CTAPOPTN contains the members that specify the installation options. Eight members are required.

- TMOSYSxx
- TMOOPTxx
- TMOSCRxx
- TMONSMxx
- TMOEDMxx
- TMODYNxx

- TMOKEYxx
- TMOSECxx

Instructions for control statements on the *hlq*.CTAPOPTN members are documented in the *Programming Guide*.

NOTE: Because TMSINIT loads the CA 1 SMF Accounting exits, you may need to have the accounting exits enabled early in the IPL process. You can have L0C6INIT execute TMSINIT directly. Include the //TMSOPTNS, //TMSRPT, //TEMPAUDT, and //TMSPARM DD statements from TMSINIT into the CAS9 proc itself. If you are using the Real-time Retention Data Set (RDS), also include the //TMSRDS DD statement.

This parameter is strictly optional. The parameter allows for TMSINIT to be executed as part of the CAS9 process itself. The functionality is not executed in a stand-alone fashion after CAS9 has completed.

If using L0C6INIT to execute TMSINIT directly, the START TMSINIT in the CAUTOCMD or COMMNDxx member in SYS1.PARMLIB is not required.

TMOSYSxx

TMOSYSxx specifies the CA 1 initialization member suffixes and is created during CA 1 installation. It is read only during initialization. The options record for TMOSYSxx identifies the member names to be overridden, if any. It contains a two-digit suffix (xx) of the number to be used in addition to the default (00). A 66-character comment field allows for user comments.

Members can be overridden by creating a new TMOSYSxx member to include the new options for TMOOPTxx, TMOSCRxx, TMONSMxx, TMOSECxx, TMOKEYxx, and TMOEDMxx. The setting is defined in TMSINIT by specifying PARM='SYS=xx'. If the options are not changing, no parameter is required.

TMOSYS00 *must* be present in *hlq*.CTAPOPTN during TMSINIT execution even if it is being overridden from the PARM option.

TMOOPTxx

TMOOPTxx specifies the system options for CA 1. TMOOPTxx contains an options record for each of the runtime options for CA 1 initialization. The options record contains the option name, which is a maximum of eight characters, and the option value, which is a maximum of 44 characters. The option value is the selected, permissible value for the option. A 24-character comment field per record is allowed for user comments. If the same option is specified multiple times, the last occurrence is used.

TMOSCRxx

TMOSCRxx specifies scratch pool definitions by naming the scratch tape pools and describing the ranges of the tapes belonging to the tape pool. The physical characteristics of the tapes for a specific scratch pool are user-defined. The contents of this member are defined through control statements that are processed by TMSINITD.

Notes:

- If TMONSMxx is used, use TMOSCRxx.
- If you intend to specify scratch subpools only using the VOL=SER= parameter, you must have at least one entry in TMONSMxx.

If specified, the table created from TMOSCRxx is used at OPEN and EOV for tapes mounted by nonspecific requests to verify that the mounted tape is from the requested scratch tape pool. The volume serial number of the tape must be in the scratch tape pool assigned. If any tape is mounted from outside the scratch pool range, CA 1 calls for another tape and issues a console message (IECTMS3) with a NOT SCRATCH 80 condition code. The tape is then demounted. Similarly, if a tape mount is satisfied by a pool-controlled volume but is not valid for the requesting data set name or job name, the volume is demounted with a NOT SCRATCH 84 condition code. It is also used by TMSCLEAN SCRATCHLIST and SCRATCHRELIST processing to report the assigned tape pool for the tape being scratched.

TMONSMxx

TMONSMxx specifies data to assign a particular scratch tape pool by a given SMS Management Class, data set name or job name. The contents of this member are defined through control statements processed by TMSINITD. For tapes mounted with nonspecific requests at OPEN and EOV, the table created from TMONSMxx is checked for the scratch tape pool to which the tape must belong to be considered valid.

Note: TMONSMxx must be used if TMOSCRxx is used.

TMOSECxx

TMOSECxx specifies internal security profiles (passwords). These internal security profiles (passwords) are used by the CA 1 online interfaces.

Note: For more information about the internal security profiles, see Chapter 11 of the *CA 1 Programming Guide*.

TMOKEYxx

TMOKEYxx allows you to modify the TMC record definitions for the ACCT field (user area) as well as to add user definitions to the BATCH_ID table and the ROBOT_TYPE table.

Note: For more information about specifications, see Chapter 7 of the *CA 1 Programming Guide*.

TMOEDMxx

TMOEDMxx contains the rules for identifying a tape as controlled by an External Data Manager (EDM). TMSINIT loads the rules into an internal table and sorts the rules in ascending sequence (most to least specific) by data set name, program name, job name and ddname. The sorted table is then loaded into common storage.You can define multiple rules with pattern matching capabilities and exits to be used in determining whether a volume is EDM controlled.

TMODYNxx

TMODYNxx specifies the CA 1 CA Dynam/T Interface options.

Realtime Expiration

CA 1 provides an optional feature known as Realtime Expiration. This enables you to apply the expiration dates defined in the Retention Data Set (RDS) to tape files as they are created in realtime. This feature is activated by adding a TMSRDS DD statement to the TMSINIT procedure that points to your installation's RDS. When TMSINIT finds the TMSRDS DD defined, it opens and reads the RDS, tabling all expiration rules in a memory resident table in ECSA (Extended Common Storage Area.) As each tape file being created goes through data set CLOSE processing, CA 1 accesses the memory resident RDS table to perform a lookup on any expiration rules matching the data set name being created. When the most specific match is made, the volume or DSNB (if a secondary file is being created) record is updated with the expiration date.

Note: With the exception of SELECT=ALL processing, Realtime Expiration uses the same matching logic as the TMSEXPDT utility. Since SELECT=ALL requires that the TMC be processed for all data sets that match the updated rule, this function may only be performed using TMSEXPDT.

When Realtime Expiration is activated on all systems in a shared TMC environment, the batch TMSEXPDT utility no longer needs to be run daily. However, TMSEXPDT continues to be supported for those users not implementing Realtime Expiration and to allow users to re-apply retention rules for tape files previously created (SELECT=ALL processing).

In multiple system or Parallel Sysplex environments, attention and care should be taken to keep the rules synchronized on all systems sharing the TMC. After changing the rules, the in-memory rules can be refreshed by simply running TMSINIT with the TMSRDS DD statement defined to point to the updated RDS. This should be done on all systems sharing the TMC to ensure that the same set of rules are applied, regardless of which system the tape file is created on.

Installations with very large Retention Data Sets should evaluate the performance of this facility prior to implementing the feature in a production environment. While the rules are accessed directly in memory, very large RDS tables may take longer to process, resulting in delays in CLOSE processing.

The Realtime Expiration feature can be deactivated by removing the TMSRDS DD statement and running TMSINIT. No system option is required to activate or deactivate this feature, however, the Retention Override parameter must be turned on to allow Realtime Expiration to override expiration dates set through JCL, as it is required for processing TMSEXPDT.

Activating CA 1

The normal operator response to the IEFTMS0 message is U to activate CA 1. Optionally, CA 1 can be deactivated by responding with the CA 1 system password instead of U. Users may initialize CA 1 with the following MVS command:

```
START TMSINIT,PRMI='parm'
```

If TMSINIT is included as part of the CAIRIM process, CA 1 is normally started as a system task by CAIRIM. Once the initialization is complete, TMSINIT ends. TMSINIT does not remain active as a task in your environment.

System software such as console directors, cross-domain managers, security packages, and so forth, should normally be initialized prior to CA 1 during a system IPL process.

Example:

START TMSINIT would be entered through the console.

Job Control Statements

```
//stepname EXEC PGM=TMSINIT[,PARM='parm']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TEMPAUDT DD  UNIT=SYSDA,
//             SPACE=(TRK,(10,1)),DCB=BLKSIZE=N*96
//TMSPARM  DD  DSN=hlq.CTAPOPTN,DISP=SHR
[//TMSRDS   DD  DSN=your.rds.data.set,DISP=SHR]
```

JCL Considerations

TMSRDS DD

Defines your retention Data Set (RDS.) This DD statement is optional and should only be specified if the Realtime Expiration feature is to be used.

Notes:

- The BLKSIZE of the TEMPAUDT DD may be any multiple of the 96 LRECL.
- If you do not have the load library for CA Common Services in the link list add a STEPLIB DD to the TMSINIT JCL that points to the CA Common Services load library.

Parameter Definitions

NODATE

Suppresses all WTORs normally issued by TMSINIT such as IEFTMS0. CA 1 is automatically activated and the current system date is used as is.

TEST

Used to validate changes to the system option members in *hlq*.CTAPOPTN. Running with this parameter does not affect the current status of CA 1.

SYS={*00*|xx}

Defines the initialization members for CA 1 where xx is a user-defined two-character alphanumeric or national (@,#,$) value. The default is 00 and refers to TMOSYS00. When using xx for initialization, TMOSYS00 must also be present.

OPT={*00*|xx|}

Specifies the system options for CA 1 where xx is a user-defined two-character alphanumeric or national (@,#,$) value. This parameter overrides the entry in SYS=xx. The default is 00 and refers to TMOOPT00.

SCR={*00*|xx|NONE}

Specifies the scratch pool names and assigned volume ranges for CA 1 where xx is a user-defined two-character alphanumeric or national (@,#,$) value. This parameter overrides the entry in SYS=xx. The default is 00 and refers to TMOSCR00. If NONE is specified for SCR, both SCR and NSM are disabled.

NSM={*00*|xx|NONE}

Specifies the scratch pool assignments for CA 1 where xx is a user-defined two-character alphanumeric or national (@,#,$) value.This parameter overrides the entry in SYS=xx. The default is 00 and refers to TMONSM00. If NONE is specified for NSM, both NSM and SCR are disabled.

EDM={*00*|xx|NONE}

Specifies the EDM rules used by CA 1 to determine whether a volume is to be EDM controlled, where xx is a user-defined two-character alphanumeric or national (@,#,$) value. This parameter overrides the entry in SYS=xx. The default is 00 and refers to TMOEDM00. If NONE is specified, EDM is disabled.

SEC={00|xx|NONE}

Specifies the Security rules used by CA 1 to determine the security profiles that should be used, where xx is a user-defined two-character alphanumeric or national (@,#,$) value. This parameter overrides the entry in the SYS=xx. The default is 00 and refers to TMOSEC00. If NONE is specified for SEC, then the default profiles are used.

KEY={00|xx|NONE}

Specifies the keywords used by CA 1 to subdivide the ACCT field, to add new BATCHIDs, and robot types, where xx is a user-defined two-character alphanumeric or national (@,#,$) value. This parameter overrides the entry in the SYS=xx. The default is 00 and refers to TMOKEY00. If NONE is specified for KEY, then no subdivision of the ACCT field is done, no new BATCHIDs or robot types are added.

DYN={00|xx|NONE}

Specifies the CA 1 CTAPOPTN member to be used for the CA Dynam/T Interface options, where xx is a user-defined two-character alphanumeric or national (@,#,$) value. This parameter overrides the entry in the SYS=xx. The default is 00 and refers to TMODYN00. If NONE is specified for DYN, then DYNam interface is disabled.

Deactivating CA 1

TMSINIT can deactivate CA 1 by responding with the CA 1 system password to the IEFTMS0 WTOR. PARM=NODATE cannot be specified.

TMSINIT (without the TEST parameter) must be used to reactivate CA 1 if it has been deactivated by using the password. The operator responds U to the IEFTMS0 message.

If *hlq*.CTAPOPTN is not present or the TMC or Audit data sets have been corrupted and TMSINIT cannot be initialized, CA 1 must be deactivated to correct the error. In this case, CA 1 can be deactivated by specifying TMSPARM DD DUMMY or by omitting TMSPARM DD from the TMSINIT JCL.

Job Control Statements

See the previous set of Job Control Statements under Activating CA 1 (see page 334).

Deactivating Realtime Expiration

To deactivate the Realtime Expiration feature you must remove the TMSRDS DD statement from the TMSINIT procedure. This deactivates the Realtime Expiration feature only on the system where TMSINIT is run. To deactivate the feature entirely in your shared TMC environment, you must run TMSINIT with the TMSRDS DD removed on every system.

Batch Activating CA 1

CA 1 can be initialized as batch active to allow testing of CA 1 batch utilities and the Online/Inquiry facility without having the CA 1 OSI installed. The operator should reply *system-password,BATCH* to the IEFTMS0 WTOR. PARM=NODATE cannot be specified.

Important! Before batch activating CA 1 you must stop all tape processing on the system. This should be used only when CA 1 is first being installed or for specified database modification jobs. Tape processing is not tracked but CA 1 batch and online facilities are functional.

Report Description

The subprogram TMSINITD produces TMS Report 00 - Initialization Report. This report contains the initialization options used during IPL, and all message intercept and scratch pool management data. A sample of this report appears on TMS Report 00 - Initialization Report (see page 338).

Note: If the Initialization Report is not available, run TMSAUDIT requesting Type 5 Audit records to generate TMS Report 32 - Audit of Functional Modifications (see page 113) listing CA 1 execution options, or execute the TMSSTATS program.

Control Statement Specification

There are no control statements for TMSINIT.

Program Return Codes

A program return code is issued based on your response to the IEFTMS0 message. The use of PARM=TEST will always result in a return code of 0. The return codes associated with various operator responses are noted below:

Return Code	Operator Response	CA 1 Status
00	U	ACTIVE
04	password,BATCH	BATCH ACTIVE
08	password	INACTIVE

TMS Report 00 - Initialization Report

```
SUNDAY, FEBRUARY 27, 2005.056                   Y O U R   C O M P A N Y   N A M E                                     ROxxxxx PAGE=00001
TMS REPORT-00          INITIALIZATION REPORT
          EXECUTION PARAMETERS RECEIVED = NODATE,OPT=00
       BEGINNING TMOOPT00 PROCEDURE

          TMOOPT00 PROCESSING COMPLETE
       BEGINNING TMOEDM00 PROCEDURE

          TMOEDM00 PROCESSING COMPLETE
       BEGINNING TMOSCR00 AND TMONSM00 PROCEDURE
          *  CREATED  95/05/13 BY CAIUSER
        1 SCRPOOL=TEMP-TAPEPOOL,
             RANGE=980000-980003
        2 SCRPOOL=OTHER-TAPES,
             RANGE=000001-000020
        3 SCRPOOL=TEST-TAPEPOOL,
             RANGE=980038-980040
        4 SCRPOOL=SMSPOOL,
             RANGE=980016-980020
        5 SCRPOOL=PATTPOOL,
             RANGE=980010-980015
          *  CREATED  95/07/27 BY CAIUSER
        1 DSN=OTHER-,
          JOB=O-,
          UCB=-,
          POOL=OTHER-TAPES
          *
        2 DSN=PATTERN.LISTING,JOB=-,POOL=PATTPOOL
          *
```

```
SUNDAY, FEBRUARY 27, 2005.056                   Y O U R   C O M P A N Y   N A M E                                 PAGE=00002

TMS REPORT-00          INITIALIZATION REPORT

        3 DSN=TESTOL#.*TAPE,
          JOB=ST-,
          UCB=E8-,
          POOL=TEST-TAPEPOOL
          *
        4 DSN=&.&TEMP.,
          JOB=-,
          UCB=E!!,
          POOL=TEMP-TAPEPOOL
          *
        5 DSN=-,
          JOB=-,
          UCB=E82,
          POOL=PATTPOOL
          *
        6 DSN=ABCDE-,
          JOB=ABCDE-,
          UCB=A-,
          POOL=TEST-TAPEPOOL
          *
        7 MGMTCLAS=INTERIM,
          POOL=SMSPOOL
          *
        8 MGMTCLAS=SMSMGMT1,
          POOL=TEST-TAPEPOOL
          *
        9 MGMTCLAS=SMSSMS,
          POOL=TEST-TAPEPOOL

          TMSINITD PROCESSING FOR CTAPOPTN MEMBER TMOSCR00 COMPLETE
          TMSINITD PROCESSING FOR CTAPOPTN MEMBER TMONSM00 COMPLETE

* ----->  IEFTMS1 CA 1(TMS) IS  ACTIVE
```

TMSINITD Utility—Analyze CAI.CTAPOPTN Scratch Pool Management Members

TMSINITD is called by TMSINIT during initialization to read the TMOSCRxx and TMONSMxx members from *hlq*.CTAPOPTN. It validates these members and produces Type 5 Audit records showing *hlq*.CTAPOPTN contents.

TMSINITD also verifies the syntax of the members and flags any errors. The internal forms of the members are returned to TMSINIT, which generates TMS Report 00 - Initialization Report. This report describes the contents of the *hlq*.CTAPOPTN members used during initialization. If presented with empty members, the completion code is set to eight.

TMSINITD may also be run in batch mode to edit the control statements for the Scratch Subpool Table and the Nonspecific Mount Rules Table. The control statements are read and edited just as they would be during initialization, but the status of existing tables in CA 1 is *not* affected. TMS Report 53 - TMSINITD Batch Report shows the control statements and any error messages.

CA 1 may be active, batch active, or inactive to run TMSINITD.

Report Description

In batch mode, TMSINITD produces TMS Report 53 -TMSINITD Batch Report. Each record read from *hlq*.CTAPOPTN is reflected on the report along with any errors and corresponding messages. A sample of this report appears on TMS Report 53 - TMSINITD Batch Report (see page 340).

Job Control Statements

The JCL to execute TMSINITD in batch mode is:

```
//stepname EXEC PGM=TMSINITD,
//              PARM='SCR=xx,NSM=xx'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TMSPARM  DD  DSN=hlq.CTAPOPTN,DISP=SHR
```

Parameter Definitions

SCR=xx

Indicates the TMOSCRxx member that is read from *hlq*.CTAPOPTN. This parameter is *required*.

NSM=xx

Indicates the TMONSMxx member that is read from *hlq*.CTAPOPTN. This parameter is *required*.

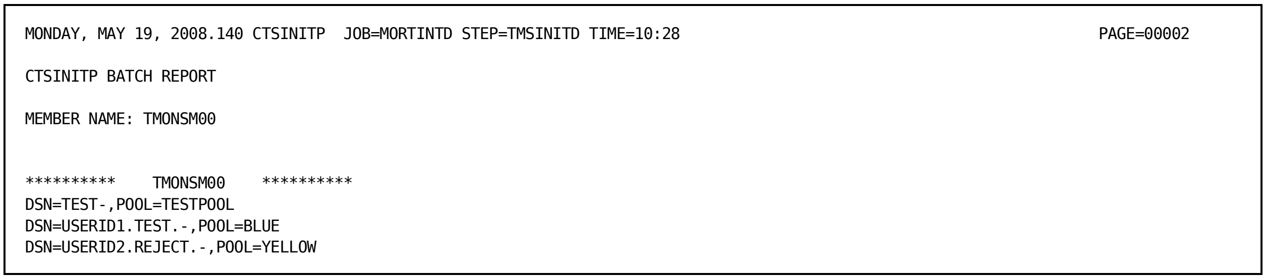
TMSINITD reads two hlq.CTAPOPTN members, TMOSCRxx and TMONSMxx:

- TMOSCRxx names the scratch tape pools and defines the ranges of the tapes belonging to the scratch tape pool.
- TMONSMxx associates data sets or job names with a particular scratch tape pool.

Note: For information on control statement syntax, see the *Programming Guide*.

TMS Report 53 - TMSINITD Batch Report

```
MONDAY, MAY 19, 2008.140  CTSINITP  JOB=MORTINTD  STEP=TMSINITD     TIME=10:28:47                                  PAGE=00001
CTSINITP BATCH REPORT

MEMBER NAME: TMOSCR00



**********    TMOSCR00    **********
SCRPOOL=BLUE,RANGE=980000-980100
SCRPOOL=YELLOW,RANGE=760110-760160
SCRPOOL=TESTPOOL,RANGE=990000-990100
```



```
MONDAY, MAY 19, 2008.140 CTSINITP  JOB=MORTINTD STEP=TMSINITD TIME=10:28                                     PAGE=00002

CTSINITP BATCH REPORT

MEMBER NAME: TMONSM00


**********    TMONSM00    **********
DSN=TEST-,POOL=TESTPOOL
DSN=USERID1.TEST.-,POOL=BLUE
DSN=USERID2.REJECT.-,POOL=YELLOW
```

TMSINITD Report Field Definitions

MEMBER NAME:

Member name processed

Control statement image

Input control statement

Program Return Codes

Possible return codes and their meanings are listed below:

Return Code	Explanation
0	Successful completion.
8	Error. When found, see TMSRPT Report - 53 for details.

TMSLBLPR Utility—Process External Tape Labels

Use the TMSLBLPR utility to generate external gummed label requests.

All label requests are processed by the Online Label Interface. For complete information on the Online Label Interface see the *Administration Guide*.

The information for the labels is derived from Audit data set records created during CA 1 realtime tape tracking. In multi-CPU environments, this allows labels to be generated from a single source for tape volumes used by systems that share the Audit data set.

TMSLBLPR periodically scans the Audit data set to determine if new records exist that indicate a label should be produced. A user exit, TMSXITL, may be called from TMSLBLPR to provide the means for creating labels on a selective basis. If this exit is not used, TMSLBLPR will produce a label request when:

- The Audit record is a Type 2 and indicates that a tape volume has been opened for output (X'00' or X'01' in FLAG1) for the primary file. DSNB records are not considered (X'FF' in the first byte of the record indicates it is a DSNB).
- The Audit record is a Type 3, and indicates an exception code of 005 or 006, which means a foreign tape volume or foreign tape multivolume data set is being created.

If the exit is invoked, it must define the type of Audit record and any other site-defined criteria that must be met for a label to be generated. Generally, this means that the record is a Type 2 Audit record. The first 340 bytes of a Type 2 Audit record that does not contain X'FF' in the first byte is the image of a TMC volume record. Any field within the record may be used to determine whether a label should be generated. See the *Programming Guide* for complete information on the coding of TMSXITL.

TMSLBLPR can be executed as a started task or as a batch program. Once it has scanned the Audit data set to the most recently created Audit record, TMSLBLPR turns itself off for a parameter-driven period of time. When this period expires, TMSLBLPR turns itself on again and picks up the Audit information from the point at which it previously ceased processing. If executed as a started task, TMSLBLPR continues to execute until the console operator issues the system STOP command. If executed as a batch program, TMSLBLPR will continue to execute until the system CANCEL command is issued.

CA 1 should be active, but may be batch active, to run TMSLBLPR.

Note: TMSLBLPR does not support a BLOCKSET mode audit data set.

Report Description

TMSLBLPR does not generate a report.

Job Control Statements

If TMSLBLPR is to be started as a system task, the following JCL should be added to SYS1.PROCLIB or to the executable procedure library of your choice.

```
//TMSLBLPR EXEC PGM=TMSLBLPR,
//             TIME=1440,PARM=',parm'
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//AUDIT    DD  DSN=CAI.CA1.AUDIT,DISP=SHR
```

Parameter Definitions

Each of the following parameters is optional and may be specified only once.

START={WARM|COLD}

'START=WARM' begins generating labels from the point where TMSLBLPR ceased during the last execution. TMSLBLPR uses the Audit data set as input. Control is based on the STOPPING POINT OF LAST EXECUTION OF TMSLBLPR data in the Audit header record.

The first external tape label generated is for the first output tape created after TMSLBLPR stopped. If the stopping point is zeros, as it is for users already under CA 1 control who have not previously executed TMSLBLPR, then PARM='START=COLD' should be used. Reformatting the Audit data set will reset the stopping point to one. 'START=WARM' is the default.

When 'START=COLD' is used, control is based on the NEXT AVAILABLE AUDIT RECORD TO BE WRITTEN in the Audit header record. The first external tape label generated is for the next output tape to be created.

WAIT={30|nnn}

Indicates a delay time period before TMSLBLPR begins. The default is 30 seconds. The value of nnn is time expressed in seconds and must be numeric. The length can be from 1 to 3 bytes. The value cannot be less than 5 or greater than 999.

EXIT={TMSXITL|modulename}

Is the module name of a user exit routine which may be used to select, reject, or modify the fields that are used to generate the label request. If no user exit module name is specified, TMSXITL is the standard default. If a user exit other than TMSXITL is used, the module must reside in either the STEPLIB or a link listed data set. An example of the user exit routine may be found in the CA 1 source library under the member name TMSXITL and is discussed in detail in the *Programming Guide*.

The following are sample parameters for this program:

```
PARM='WAIT=25'
```

```
PARM='START=WARM'
```

```
PARM='WAIT=67,START=COLD'
```

Control Statement Specification

There are no control statements for TMSLBLPR.

TMSMERGE Utility—Merge TMC Data

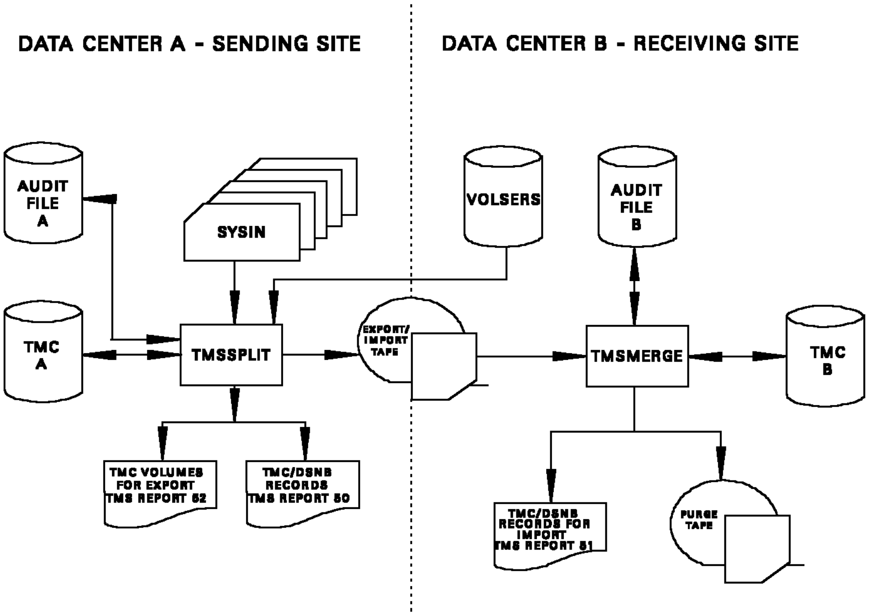
Use the TMSMERGE utility to aid data center mergers or decentralization.

This utility performs the second half of a CA 1 split/merge operation by reading an import tape containing information from a sending TMC. This input is created by the TMSSPLIT utility discussed on TMSSPLIT (see page 396).

Normally, TMC Volume records from the tape overlay the record of the same volume in the receiving (active) TMC after a copy of the existing record is written to a *purge* data set. If the incoming Volume records have associated DSNBs, unused DSNBs in the receiving TMC are allocated after the Volume record is established. The figure on the next page illustrates the TMC Split/Merge operation. However, if the AIVS= control statement is used, no active TMC records are overlaid. Instead, each volume record read from the import file is assigned a new external VOLSER that is currently in delete status. This requires enough volume records to be in delete status to receive the entire import file. When a new VOLSER is assigned, it is no longer in delete status, so RESTART cannot be used if an abend should occur when using the AIVS= control statement.

CA 1 must be active or batch active to run TMSMERGE. Batch active is the recommended state.

The following diagram illustrates the TMC Split/Merge:


DATA CENTER A - SENDING SITE
DATA CENTER B - RECEIVING SITE
AUDIT FILE A
SYSIN
TMC A
TMSSPLIT
EXPORT/ IMPORT TAPE
TMC VOLUMES FOR EXPORT TMS REPORT 52
TMC/DSNB RECORDS TMS REPORT 50
VOLSERS
AUDIT FILE B
TMSMERGE
TMC B
TMC/DSNB RECORDS FOR IMPORT TMS REPORT 51
PURGE TAPE

Report Description

TMSMERGE produces TMS Report 51 - TMC/DSNB Records for Import. The first page of the report provides Audit and TMC DSNB record requirements. The second page lists the Volume record and DSNB record information to be merged in volume serial number sequence. Other information reported includes the volume sequence number, file sequence number, data set name, creating job name, creation date and any error message. A sample of this report appears on TMS Report 51 - TMC/DSNB Records for Import (see page 350).

Considerations

The receiving site should take the following into consideration:

- The use of Catalog Control and Catalog Days Control of tape data sets (CATALOG, CATLG, or CATLG/ddd in control statements and online interfaces, or LABEL=EXPDT=99000 and LABEL=EXPDT=90ddd in JCL) indicates that CA 1 keeps tapes out of scratch status as long as the CATLG-protected data set has an entry in the system catalog. If the system catalog information from the sending site is not immediately available to the receiving site, the batch utility TMSCTLG should *not* be executed until the system catalog is updated or the merged TMC records are updated with some other EXPDT value.
- If OCTLG=NO in *hlq*.CTAPOPTN member TMOOPTxx, run TMSOSCAT PARM=SYNC again if the receiving data is from a Version 5.2 system or below that has not had TMSOSCAT PARM=SYNC run against the data.
- If Volume records from the sending site are under Vault Management System control, the Vault Pattern entries from the sending site should be integrated into the receiving site's pattern prior to running any VMS utilities. If the sending site uses vault names (OUTCODEs) equal to vault names used in the receiving site, the sending site's vault pattern entries need to have the REDEFINED keyword option specified for each data set name affected to prevent duplicate slot errors.
- If the sending or receiving site uses the Scratch Pool Management feature, the hlq.CTAPOPTN members defining the tape pool options need to be built/integrated. Console operators and users should be informed of any visible changes this may cause. The same holds true for the CA 1 options defined in the TMOOPTxx and TMOEDMxx members of hlq.CTAPOPTN. The Retention Data Set used by TMSEXPDT, and the PDS used by TMSPULL, will require similar accommodation if either data set is in use at either location.

  Note: When using PARM=DEFAULT with TMSMERGE, the FLAG1 =x’01’ ('eligible for RDS override') bit will be turned on in ALL the new merged TMC records. If there is a matching RDS entry, the EXPDT of the tape is overridden. To ensure that the proper expiration date is assigned to your datasets: review your RDS entries or incorporate your sending RDS entries into the receiving site’s Retention Dataset (RDS) before running TMSEXPDT. Otherwise, premature retention expiration may occur.

- A split/merge operation should not be performed when the sending site's ranges are identical to ranges in the receiving site's TMC that currently reflect nonscratch data sets unless new external VOLSERs are assigned. The merge operation requires that the Volume records for the import data set be properly formatted in the receiving TMC, and they should be in either scratch or inactive (DELETE) status. If TMSMERGE determines that you are overlaying an active record, it will free the DSNBs and break the volume chains and then do the overlay.
- The number of Audit records required to support the merge is specified on the TMSSPLIT output. This number is to be combined with the value of the number of Audit and DSNB records reported. The number is also relative to the Audit data set threshold level; should the level be reached TMSMERGE will terminate with a U0008 abend. Another consideration is that more than the number specified by TMSSPLIT is required if the merge overlays records that are active; that is, records not in scratch or delete status, especially if the volume records being overlayed are multifile volumes and the DSNBs associated with them have to be cleared and returned to the free chain by the merge process. Since the number of audit records needed if AIVS or foreign control statements are used is increased, plan for an additional audit record for each secondary volume and secondary VOLSER. A good practice would be to triple the number of AUDIT records based on the output from TMSSPLIT.

  Because RESTART cannot be used if AIVS or foreign control statements are used, the only way to recover from an abend situation is to restore the TMC. For this reason, CA 1 should be in batch active mode. It is recommended that a TEST parameter be used for the initial run to ensure there are no problems prior to the production run.

Merge Procedure (without using AIVS or foreign control statements)

Step 1

Execute TMSBINQ with the HEADERS control statement to generate a report on the TMC and Audit Control Records. The output of TMSSPLIT (TMS Report 50) shows the number of DSNB and Audit records required of the receiving site. The output of TMSBINQ provides verification that sufficient records exist in each database for the merge operation. The output of TMSSPLIT also shows each Volume record that is to be placed in the receiving TMC.

If the receiving TMC does not have the necessary records available, TMSFORMT needs to be executed to perform an extend operation and add any additional Volume or DSNB records as required. If the Audit data set does not contain the required number of records identified by TMSSPLIT, TMSFORMT needs to be executed to reformat the Audit data set. (It can later be reformatted to its original size.) For additional information, see Extending the TMC (see page 265) and Formatting the Audit Data Set (see page 262).

Step 2

Execute TMSPTRS on the receiving TMC and correct any multivolume or multi-data set chaining errors reported. TMSPTRS must run with a condition code of 0 before you continue to the next step.

Step 3

TMSMERGE requires that CA 1 is initialized into either a batch active or active status. While batch active, CA 1 will not track any realtime tape activity but a nominal amount of such activity is indicated if CA 1 remains active. No other CA 1 batch or online processing should be performed during the merge, whether CA 1 is active or batch active.

Step 4

Execute TMSCOPY to back up the receiving TMC and reset the Audit data set pointers.

Step 5

Execute TMSMERGE using the import tape created by TMSSPLIT.

Step 6

Review the reports generated by TMSMERGE. In a successful merge, the output of TMSMERGE should match that of TMSSPLIT. If desired, the receiving TMC can be backed up with TMSCOPY.

Step 7

If the merge is invalid (for example, a Volume record was unintentionally overlaid), the TMC can be recreated with TMSMERGE. Modify the IMPORT DD to reference the purge file created in the bad merge and catalog a new PURGE DD file. Once a corrected import tape is available, the merge procedure can be reexecuted.

Note: As an alternative, the TMC can be restored by executing the TMSCOPY utility with PARM='RESTORE,NOAUDIT' using the backup tape created in STEP 4.

Step 8

If the merge abended and TMC recreation is not required, the PARM=RESTART option directs TMSMERGE to compare the sending TMC records with the receiving TMC. No Audit records are created until the first discrepancy in the TMC records is encountered, at which point TMSMERGE resumes the update process.

Step 9

Initialize CA 1 as active if CA 1 was made batch active in Step 3.

Job Control Statements

```
//stepname EXEC PGM=TMSMERGE
//              [,PARM='[DATEFMT=(fmt)][,TEST][,RESTART][,DEFAULT][,FAST]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//IMPORT   DD  DSN=CAI.CA1.EXPORT,DISP=(OLD,KEEP),UNIT=TAPE
//PURGE    DD  DSN=CAI.CA1.PURGE,DISP=(NEW,KEEP),UNIT=TAPE,
//             DCB=BLKSIZE=n*366                                *NOTE
//SYSIN    DD  *
(control statements here)
/*
```

Note: The BLKSIZE may be any multiple of the LRECL, 366. If no BLKSIZE is specified, the BLKSIZE for PURGE defaults to 32574 for tape or 8784 for DASD devices.

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 51. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21)

TEST

Prevents actual updating of the receiving TMC.

RESTART

Compares receiving TMC records with sending records until the first discrepancy. Begins updating at this point.

DEFAULT

Turns on the 'eligible for RDS override' (Flag1=x’01’) bit for new TMC records being merged into the TMC. If there is a matching RDS rule, this flag allows MSEXPDT to override the EXPDT currently on the tape. Review your RDS rules to verify that the new EXPDT is applied to the tape files that are merged into the existing TMC. Because the RDS override indicator is turned on for all merged files, you must define all file rules in the RDS.

FAST

Eliminates the first pass of the import data set and significantly reduces processing time. However, you *must* ensure that enough free DSNB and Audit records exist for TMSMERGE to process both import TMC record and target TMC record updates. Without this PARM, TMSMERGE determines if enough DSNB and Audit records exist in the first pass before the actual updates are made in the second pass. This PARM is not valid with PARM=TEST.

JCL Considerations

TMSRPT DD

Refers to the merge report.

PURGE DD

Refers to the before image of the TMC records updated on the receiving TMC. If the restart is in progress, the disposition should be changed to MOD and the volume serial number added.

Control Statement Specification

The control statements are optional, even the SYSIN DD statement is optional. If SYSIN is present, all control statements must reside on one control statement starting at position 1, and must be separated by commas.

AIVS=xxxxxx

This specifies that **all** volume records read from the import file are reassigned a new (unique) external VOLSER starting with volume xxxxxx. Only volume records in delete status are selected for use. All volume chain information and DSNB volume pointers are changed to reflect the new external VOLSER. However, if the merging VOLSER is in the TMC and in delete status, the merging VOLSER is used instead of assigning a new AIVS VOLSER in the range specified. The Report-51 created by TMSMERGE will list the new external VOLSER in the VOLSER field. The old internal VOLSER and old external VOLSER are listed in the same line.

FOREIGN=xxxxxx

This is similar to AIVS= but only applies to volumes that have the TMNRS indicator in TMC FLAG4 X'04' turned on (indicating they are foreign volumes). A new external VOLSER is assigned, starting with the first volume in delete status after xxxxxx. The Report-51 created by TMSMERGE will list the new external VOLSER in the VOLSER field. The old internal VOLSER and old external VOLSER are listed in the same line.

CATALOG=NEVER|AIVS|ALWAYS

This control statement indicates if any recatalog operations should take place. If NEVER is specified, then no recataloging is done (this is the default). If AIVS is specified, then any AIVS or foreign data set that is assigned a new external VOLSER may be recataloged to the new external VOLSER. This recatalog operation will occur if either A) the file is currently cataloged to the old external VOLSER or B) the file is currently not cataloged on the system *and* that the OSCAT indicator is on (indicating the file had been cataloged on the system it was 'split' from). If ALWAYS is specified, a catalog is attempted for all files MERGEd that have the OSCAT indicator on. If ALWAYS is specified, files assigned a new external VOLSER are treated the same as if CATALOG=AIVS had been specified.

TMS Report 51 - TMC/DSNB Records for Import

The first page of TMSMERGE output reflects the Audit and TMC DSNB record requirements. The second page lists the data sets that are to be merged.

```
MONDAY, JUNE 27, 2005.177                 Y O U R   C O M P A N Y   N A M E                          ROxxxxx PAGE=00001

TMS REPORT-51 TMC/DSNB RECORDS FOR IMPORT


  0054301 RECORDS AVAILABLE ON AUDIT
  0050000 DSNBS ALLOCATED  0018716 DSNBS USED


  0022733 AUDIT RECORDS REQUIRED FOR IMPORT
  0002858 DSNB RECORDS REQUIRED FOR IMPORT
```

```
MONDAY, JUNE 27, 2005.177                 Y O U R   C O M P A N Y   N A M E                              PAGE=00002

TMS REPORT-51 TMC/DSNB RECORDS FOR IMPORT

  VOLSER  VSEQ  FSEQ  DATA SET NAME                           JOBNAME   CDATE

  100001  001   001   ADMIN.PAYR.DLY.G0406V00                 ADMINPDY  MMMDD YYYY  WARNING: ACTIVE TARGET RECORD OVERLAID
  100002  001   001   ADMIN.PAYR.WLY.G0088V00                 ADMINPWY  MMMDD YYYY  WARNING: ACTIVE TARGET RECORD OVERLAID
  100003  001   001   ADMIN.PAYR.DLY.G0405V00                 ADMINPDY  MMMDD YYYY  WARNING: ACTIVE TARGET RECORD OVERLAID
  100004  001   001   ADMIN.PAYR.WLY.G0087V00                 ADMINPWY  MMMDD YYYY  WARNING: ACTIVE TARGET RECORD OVERLAID
  100005  001   001   ADMIN.PAYR.DLY.G0404V00                 ADMINPDY  MMMDD YYYY  WARNING: ACTIVE TARGET RECORD OVERLAID
  100007  001   001   ADMIN.PAYR.DLYBKUP.G0403V00            ADMINPDY  MMMDD YYYY  WARNING: ACTIVE TARGET RECORD OVERLAID
  100009  001   001   ADMIN.PAYR.DLY.G0402V00                 ADMINPDY  MMMDD YYYY  WARNING: ACTIVE TARGET RECORD OVERLAID
  100010  001   001   ADMIN.PAYR.WLY.G0086V00                 ADMINPWY  MMMDD YYYY  WARNING: ACTIVE TARGET RECORD OVERLAID


  0022733 AUDIT RECORDS REQUIRED FOR RESTORING FROM THE PURGE TAPE
  0002858  DSNB RECORDS REQUIRED FOR RESTORING FROM THE PURGE TAPE
```

TMSMERGE Report Field Definitions

RECORDS AVAILABLE ON AUDIT

Number of Audit records available in Audit

DSNBS ALLOCATED

Number of DSNB records formatted in TMC

DSNBS USED

Number of used DSNB records in TMC

AUDIT RECORDS REQUIRED FOR IMPORT

Estimated number of Audit records required for import

DSNB RECORDS REQUIRED FOR IMPORT

Number of DSNB records required for import

VOLSER

Volume serial number of tape volume or file

VSEQ

Volume sequence number of tape volume

FSEQ

File sequence number of file

DATA SET NAME

Data set name of tape volume or file

JOBNAME

Creating job name of tape volume or file

CDATE

Creation date of tape volume or file

Message

Processing message

TMSMULV Utility—Construct Multivolume Pointers for TMC Volume Records

Use the TMSMULV utility when initially constructing a TMC or to accommodate off-site processing.

Its purpose is to build the forward and backward volume pointers for multivolume data sets, since no volume chaining is performed when the TMC is originally created.

The program selects Volume records that are not in inactive (DELETE) or scratch status, contain a valid volume serial number, and have a first volume pointer (1STVOL) field equal to hexadecimal zeros. They are sorted by data set name, creation date, creation time and volume sequence. Chains are then constructed for multivolume data sets, where possible. The creation date and time of the first volume for the data set are carried over into the secondary Volume records.

TMSMULV may be executed as many times as necessary to complete the chaining process. TMSMULV partially completes a multivolume chain when the identification of a secondary volume cannot be resolved with the primary volume. The Volume Chain Generation Report and the Volume Chain Error Report created by TMSMULV should be examined, and data set names appearing on both reports should be checked to ensure that volumes on the error list are not included as a part of the volume chain in the generation report.

TMSUPDTE, CA 1 ISPF or TIQ Online Inquiry/Update can be used to correct any fields and to reset the 1STVOL pointer to hexadecimal zeros in a partially constructed chain. TMSMULV can be executed again to build the proper chains.

CA 1 must be active or batch active to run TMSMULV.

Important! This program should be used only by personnel responsible for maintaining CA 1.

Report Description

TMSMULV produces TMS Report 33 - Volume Chain Generation Report and TMS Report 34 - Volume Chain Error Report.

TMS Report 33 - Volume Chain Generation Report is sorted by data set name, creation date and time, volume sequence number and volume serial number. Other information reported includes previous volume, next volume, and first volume of a multivolume chain. Summary figures are printed for the total number of records read, number of records updated and number of data set chains built. A sample of this report appears on TMS Report 33 - Volume Chain Generation Report (see page 354).

TMS Report 34 - Volume Chain Error Report contains the data set name, volume sequence number, volume serial number and type of error. A count of the errors detected is shown at the end of the report. A sample of this report appears on TMS Report 34 - Volume Chain Error Report (see page 355).

Job Control Statements

```
//stepname EXEC PGM=TMSMULV[,PARM='[DATEFMT=(fmt)][,TEST]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TMSERR   DD  SYSOUT=A
//SYSOUT   DD  SYSOUT=A
//SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK06 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
```

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Reports 33 and 34. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

TEST

No updates are made to the TMC. This parameter is optional.

Control Statement Specification

There are no control statements for TMSMULV.

TMS Report 33 - Volume Chain Generation Report

```
WEDNESDAY, JUNE 29, 2005.179                     Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-33.                                   VOLUME CHAIN GENERATION REPORT

    DATA SET NAME                           VOLUME   VOLUME    PREVIOUS    NEXT     FIRST         DATE           TIME
                                            SEQ NO   SER NO     VOLUME    VOLUME    VOLUME

DASD.BKUP.A00100.C2                          0001    500092               500218    500092     MMMDD YYYY        01.27
DASD.BKUP.A00100.C2                          0002    500218     500092    500068    500092     MMMDD YYYY        01.27
DASD.BKUP.A00100.C2                          0003    500068     500218              500092     MMMDD YYYY        01.27
DASD.BKUP.A00101.C2                          0001    500154               502020    500154     MMMDD YYYY        01.35
DASD.BKUP.A00101.C2                          0002    502020     500154    501176    500154     MMMDD YYYY        01.35
DASD.BKUP.A00101.C2                          0003    501176     502020              500154     MMMDD YYYY        01.35
DASD.BKUP.A00102.C2                          0001    500334               500782    500334     MMMDD YYYY        01.42
DASD.BKUP.A00102.C2                          0002    500782     500334    501022    500334     MMMDD YYYY        01.42
DASD.BKUP.A00102.C2                          0003    501022     500782              500334     MMMDD YYYY        01.42
DASD.BKUP.A00103.C2                          0001    502856               500082    502856     MMMDD YYYY        01.54
DASD.BKUP.A00103.C2                          0002    500082     502856    500308    502856     MMMDD YYYY        01.54
DASD.BKUP.A00103.C2                          0003    500308     500082              502856     MMMDD YYYY        01.54
DASD.BKUP.A00104.C2                          0001    502580               500986    502580     MMMDD YYYY        02.03
DASD.BKUP.A00104.C2                          0002    500986     502580    501828    502580     MMMDD YYYY        02.03
DASD.BKUP.A00104.C2                          0003    501828     500986              502580     MMMDD YYYY        02.03
DASD.BKUP.A00105.C2                          0001    502642               500378    502642     MMMDD YYYY        02.12
DASD.BKUP.A00105.C2                          0002    500378     502642    502008    502642     MMMDD YYYY        02.12
DASD.BKUP.A00105.C2                          0003    502008     500378              502642     MMMDD YYYY        02.12
DASD.BKUP.A00106.C2                          0001    500998               500050    500998     MMMDD YYYY        02.20
DASD.BKUP.A00106.C2                          0002    500050     500998    500204    500998     MMMDD YYYY        02.20
DASD.BKUP.A00106.C2                          0003    500204     500050              500998     MMMDD YYYY        02.20
DASD.BKUP.A00107.C2                          0001    500008               502870    500008     MMMDD YYYY        02.29
DASD.BKUP.A00107.C2                          0002    502870     500008    501470    500008     MMMDD YYYY        02.29
DASD.BKUP.A00107.C2                          0003    501470     502870              500008     MMMDD YYYY        02.29
DASD.BKUP.A00108.C2                          0001    501264               501996    501264     MMMDD YYYY        02.38
DASD.BKUP.A00108.C2                          0002    501996     501264    500910    501264     MMMDD YYYY        02.38
DASD.BKUP.A00108.C2                          0003    500910     501996              501264     MMMDD YYYY        02.38
DASD.BKUP.B00100.C2                          0001    501092               501218    501092     MMMDD YYYY        01.27
DASD.BKUP.B00100.C2                          0002    501218     501092    501068    501092     MMMDD YYYY        01.27
DASD.BKUP.B00100.C2                          0003    501068     501218              501092     MMMDD YYYY        01.27
DASD.BKUP.B00101.C2                          0001    501154               502120    501154     MMMDD YYYY        01.35
DASD.BKUP.B00101.C2                          0002    502120     501154    502176    501154     MMMDD YYYY        01.35
DASD.BKUP.B00101.C2                          0003    502176     502120              501154     MMMDD YYYY        01.35
DASD.BKUP.B00102.C2                          0001    501334               502782    501334     MMMDD YYYY        01.42
DASD.BKUP.B00102.C2                          0002    502782     501334    502022    501334     MMMDD YYYY        01.42
DASD.BKUP.B00102.C2                          0003    502022     502782              501334     MMMDD YYYY        01.42
DASD.BKUP.B00103.C2                          0001    502156               502082    502156     MMMDD YYYY        01.54
DASD.BKUP.B00103.C2                          0002    502082     502156    501308    502156     MMMDD YYYY        01.54
DASD.BKUP.B00103.C2                          0003    501308     502082              502156     MMMDD YYYY        01.54
DASD.BKUP.B00104.C2                          0001    501580               501986    501580     MMMDD YYYY        02.03
DASD.BKUP.B00104.C2                          0002    501986     501580    501328    501580     MMMDD YYYY        02.03
DASD.BKUP.B00104.C2                          0003    501328     501986              501580     MMMDD YYYY        02.03
DASD.BKUP.B00105.C2                          0001    501642               501378    501642     MMMDD YYYY        02.12
DASD.BKUP.B00105.C2                          0002    501378     501642    502118    501642     MMMDD YYYY        02.12
DASD.BKUP.B00105.C2                          0003    502118     501378              501642     MMMDD YYYY        02.12
DASD.BKUP.B00106.C2                          0001    501998               502050    501998     MMMDD YYYY        02.20
DASD.BKUP.B00106.C2                          0002    502050     501998    501204    501998     MMMDD YYYY        02.20
DASD.BKUP.B00106.C2                          0003    501204     502050              501998     MMMDD YYYY        02.20
```

TMS Report 34 - Volume Chain Error Report

```
TUESDAY, JULY 25, 2005.207                       Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-34.                                   VOLUME CHAIN ERROR REPORT

       DATA  SET  NAME                          VOLUME    VOLUME      ERROR
                                                SEQ NO    SER NO

SYSPROG.TEST.MVOL                               000003    100216    NON-CONSECUTIVE SEQ. NOS.
SYSPROG.TEST.MVOL                               000004    100227    NON-CONSECUTIVE SEQ. NOS.
SYSPROG.TEST.MVOLM                              000004    100262    NON-CONSECUTIVE SEQ. NOS.
SYSPROG.TEST.MVOLTST                            000002    100237    NON-CONSECUTIVE SEQ. NOS.
SYSPROG.TEST.MVOLTST                            000003    100239    NON-CONSECUTIVE SEQ. NOS.
SYSPROG.TEST.MVOLTST                            000004    100241    NON-CONSECUTIVE SEQ. NOS.

         END OF REPORT 34  -  TOTAL ITEMS LISTED=        6
```

TMSMULV Report Field Definitions

DATA SET NAME

Data set name of tape volume

VOLUME SEQ NO

Volume sequence number of tape volume

VOLUME SER NO

Volume serial number of tape volume

PREVIOUS VOLUME

Previous tape volume of multivolume chain

NEXT VOLUME

Next tape volume of multivolume chain

FIRST VOLUME

First tape volume of multivolume chain

DATE

Creation date of tape volume

TIME

Creation time of tape volume

ERROR

Error message

TOTAL NUMBER OF RECORDS READ -

Number of Volume records processed

TOTAL NUMBER OF RECORDS UPDATED -

Number of Volume records updated

TOTAL NUMBER OF DATA SET CHAINS

Number of multivolume chains

TMSOSCAT Utility—Report on TMC/Operating System Catalog Discrepancies

Use the TMSOSCAT utility to determine the differences between the TMC and the operating system catalog.

The catalogs to be compared are sorted by data set name or volume serial number, depending on the RS or SEQUENCE option. TMSOSCAT can report on tapes controlled by EDM. TMSOSCAT bypasses tapes owned by CA Dynam/T.

To compare the ICF catalog to the TMC, an IDCAMS step is added to the procedure to list all of your catalogs. The output from IDCAMS is used as input to the TMSOSCAT step in the IDCAMS DD.

Four fields are compared from each database: data set name, VOLSER, volume sequence and file sequence. An additional check for matching data set names can be performed by specifying the LOCATE=YES control statement. This allows TMSOSCAT to determine if a data set not found on the catalog listing was posted to the system catalog after the list was generated. TMSOSCAT treats any file cataloged with a file sequence of zero the same as file sequence of one.

The TMC and system catalog do not have to be synchronized to perform tape processing, but synchronization will prevent error conditions from occurring, such as the system catalog not reflecting the content of the tape header label or conflicts between the TMC and system catalog, which result in system or CA 1 abends.

In addition to reporting on the difference between the TMC and the operating system catalogs, TMSOSCAT can also be used to ensure the new TMISCAT in TMC FLAG4 X'08' and DSNBISCA in DSNB FLAG1 X'08' bits are set on for all files currently cataloged. This usage of TMSOSCAT should be done once on each system sometime after upgrading to r12. Until this has been done on all systems sharing the TMC, the new TMOOPT option, OCTLG, should be left as YES (the default). Once TMSOSCAT has been run to SYNC 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 input is accepted and the SYSIN control statements are ignored as well. Simply include a parameter of SYNC or SYNC,TEST. If TEST is included, no actual updates are performed, but the total line stating how many TMC records *would* have been updated is produced. If SYNC is used without TEST, then the total line states how many TMC records *were* updated.

CA 1 must be active or batch active to run TMSOSCAT.

User Exit

User exit TMSXOCAT is provided for tailoring TMSOSCAT processing. It is invoked after each TMC record is selected for processing and is called only if the option UXOCAT is set to the name of the exit itself in the TMOOPTxx member of *hlq*.CTAPOPTN.

TMSXOCAT can continue processing or reject the record.

Note: For more information on this user exit, see the *Programming Guide*.

This exit is not used when PARM=SYNC is specified.

Report Description

TMSOSCAT generates TMS Report 17 - Tape Management Catalog (TMC) and MVS Catalog Differences. 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 date set name, volume serial number, volume sequence number, file sequence number, expiration date and an error code. System catalog fields reported include date 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.

Samples of this report are provided, beginning on TMS Report 17 - Tape Management Catalog and Operating System Catalog Differences (see page 362).

Job Control Statements

The IDCAMS step is executed to list the ICF catalog entries.

```
//IDCAMS   EXEC PGM=IDCAMS
//SYSPRINT DD  DSN=&.&IDCAMS.,DISP=(NEW,PASS),UNIT=SYSDA,
//             SPACE=(CYL,5)
//SYSIN    DD  *
 LISTCAT NONVSAM ALL                                     Notes
 LISTCAT NONVSAM ALL CATALOG(UserCatalogDataSetName)     Notes
//TMSOSCAT EXEC PGM=TMSOSCAT[,PARM='parm']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//IDCAMS   DD  DSN=&.&IDCAMS.,DISP=(OLD,DELETE)
//SYSOUT   DD  SYSOUT=A
//SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK06 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SYSIN    DD  *
 (control statements here)
/*
```

Note:

1. This IDCAMS control statement lists only the system master catalog.
2. Code this IDCAMS control statement for each user catalog thatcontains the tape data set names controlled by CA 1.

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 17. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

SYNC

Specifies that all active files (not scratch and not deleted) in the TMC should be checked with the operating system catalogs to see if they are currently cataloged. If they are, the new TMISCAT and DSNBISCA bits are set.

RESYNC

Specifies that all active files on volumes (that are not in scratch or deleted status) in the TMC, should be checked with the operating system catalogs to see if they are currently cataloged. If the catalog indicators in the TMC record match the status in the OS Catalog (both say the file is NOT currently cataloged or both say the file IS currently cataloged), then no information is printed in the report and no updates are performed.

If the DSN being checked is found in the OS Catalog and the TMC catalog indicators are not currently on the following happens:

- The message "DSN FOUND ON CATALOG=" is printed in the report.
- The catalog indicators (TMISCAT, TMCATLOG, DSNBISCA, and DSNBWSCA) are set ON.
- The catalog-counter fields (TMCATCNT and DSNBCCNT) are set to a value of 1.

If the DSN being checked is *not* found in the OS Catalog and the TMC catalog indicators are on the following happens:

- The message "DSN –NOT- ON CATALOG=" is printed in the report.
- The current catalog indicators (TMISCAT and DSNBISCA) are turned off.
- The catalog-counter fields are set to a value of 0.

If your z/OS environment is setup where the CA 1 TMC is shared by systems that do **not** share all MVS/Catalogs and you run TMSOSCAT with PARM=RESYNC on one system; then you **must** run TMSOSCAT with PARM=SYNC on all other systems. A file that is cataloged in an MVS/Catalog and is not available on the system where TMSOSCAT was run with PARM=RESYNC has its catalog indicators turned off. You must run TMSOSCAT with PARM=SYNC on the other system or systems to turn the indicators back on.

If your z/OS environment is set up whereas all systems sharing the TMC also share the same MVS/Catalog structure (all user catalogs are shared and absolutely no tape data sets are cataloged in any unique master catalogs;) then simply running TMSOSCAT with PARM=RESYNC on one system is sufficient.

TEST

Used if SYNC or RESYNC was specified. The only difference is that the actual update of the TMC records are bypassed. The total count reflects how many records would have been updated.

DETAIL

Specified that a detail line in TMS Report 17 should be printed for each file being updated, or that would have been updated if TEST is also specified.

JCL Considerations

IDCAMS

Defines the SYSPRINT output from an IDCAMS ICF catalog list. The VSAM catalog entries are not analyzed. The SYSIN control statement *must* be LISTCAT NONVSAM ALL. If multiple user catalogs are to be processed, the IDCAMS output should be MODed to a single data set.

Control Statement Specification

The control statements for TMSOSCAT are free-form and cannot exceed position 71. Additional records are added to continue a control statement.

The keyword and its operand must be contained in one control statement. Each set of keywords and operands on a control statement are separated by a comma except for the last keyword and operand which is delimited by a blank. The format for the control statement is:

```
keyword=value,...,keyword=value
```

Control Statement Definition

V={*NO*|YES} or VALID={*NO*|YES}

YES specifies only unexpired records that are not in scratch or delete status. NO is the default and causes all active (nondelete status) records to be processed.

E={*NO*|YES} or EXPCATLG={*NO*|YES}

YES specifies only TMC records under Catalog Control are to be processed. If EXPCATLG=YES is specified, VALID=YES is assumed. NO is the default.

Note: The EXPCATLG and EDM control statements are mutually exclusive.

I={*NO*|YES} or ISCAT={*NO*|YES}

YES specifies that only TMC records that have the TMISCAT FLAG4 X'08' (DSNBISCA FLAG1 X'08' for DSNBs) indicator on are processed. If I=YES is specified, VALID=YES is assumed. ISCAT is mutually exclusive with EXPCATLG, SCRTCH, and EDM.

C={DAYS=nnnn|DATE=date} or CDATE={DAYS=nnnn|DATE=date}

Specifies only TMC records with a creation date greater than CDATE.

DAYS=nnnn

Is the number of days from the current date. Four digits must be specified.

DATE=date

Is the actual date. Delimiters are required if the actual date contains blanks or commas.

S={*NO*|YES} or SCRTCH={*NO*|YES}

YES specifies only active (nondelete status), scratched TMC records are to be processed. SCRTCH=YES (S=YES) is mutually exclusive with VALID=YES and EXPCATLG=YES. NO is the default.

N=(value1,value2,...) or NODE=(value1,value2,...)

Specifies only TMC and system catalog records with nodes (high-level index) matching one of the nodes in the value list are to be processed. Each node is specified as one to eight characters. A maximum of eight nodes may be specified.

L={*NO*|YES} or LOCATE={*NO*|YES}

An entry is made on an internal list of the data sets found in the TMC and system catalog which do not match. LOCATE=YES (L=YES) issues a LOCATE to the MVS catalog for each entry on the list. If LOCATE=NO (L=NO) is specified, a LOCATE is not issued. NO is the default.

RS={*DSN*|VSN} or SEQUENCE={*DSN*|VSN}

Determines the report sequence.

DSN

Produces a report in data set name sequence. Specifying DSN means that you consider the entries in the system catalog to be correct. DSN is the default.

VSN

Produces a report in volume serial number sequence. Specifying VSN means that you consider the entries in the TMC to be correct.

RC={TOTALS|ERRORS} or CONTENTS={TOTALS|ERRORS}

Determines the report contents.

TOTALS

Produces a report containing totals only, instead of complete detail with totals.

ERRORS

Produces a report containing error detail and totals.

Note: If RC= is not specified, the report is a detailed list of all files selected, with or without errors.

X={NO|YES} or EDM={NO|YES}

Indicates whether tapes controlled by EDM are to be considered by the program.

NO

Indicates that EDM tapes are to be included.

YES

Indicates that EDM tapes are to be excluded.

Default: NO

Note: The EXPCATLG and EDM control statements are mutually exclusive.

$

Marks the end of a set of control statements and the start of another set. Must be in position 1. More than one set of control statements may be in SYSIN. For each set, TMSOSCAT sets the options specified by the set and processes the catalogs.

At the end of processing for a set, all options are reset. The last set should not be terminated by a $.

Control Statement Example

Two data set nodes (CA1RAY and CA1ROB) were accidentally deleted from the system catalog. All tape data sets created by these applications use Catalog Control (EXPDT=99000). TMSCTLG has not been executed since the accidental deletion. The control statement for TMSOSCAT is:

```
EXPCATLG=YES,NODE=(CA1RAY,CA1ROB),RS=VSN
```

This input generates a listing of all the TMC records for data sets starting with CA1RAY and CA1ROB that are under Catalog Control.

TMS Report 17 - Tape Management Catalog and Operating System Catalog Differences

Example 1

```
THURSDAY, FEBRUARY 24, 2005.060                 Y O U R   C O M P A N Y   N A M E                            ROxxxxx PAGE=00001
TMS REPORT-17 TAPE MANAGEMENT CATALOG (TMC) AND OS/VS CATALOG DIFFERENCES
          DATA SET NAME (TMC)                VSN   VSQ FSQ   EXPDT             DATA SET NAME (CATALOG)             VSN

                              I=YES,RS=DSN,RC=TOTALS,L=YES
```

```
THURSDAY, FEBRUARY 24, 2005.060                  Y O U R   C O M P A N Y   N A M E                              PAGE=00002
 TMS REPORT-17 TAPE MANAGEMENT CATALOG (TMC) AND OS/VS CATALOG DIFFERENCES
           DATA SET NAME (TMC)                VSN   VSQ FSQ   EXPDT             DATA SET NAME (CATALOG)             VSN

                                         ERROR COUNT TOTALS -   DESCRIPTION
                                         CODE 1           0   -   DATA SET NAME DIFFERS (VSN SEQ ONLY)
                                         CODE 2           1   -   VOLUME SERIAL NUMBER DIFFERS (DSN SEQ ONLY)
                                         CODE 3           0   -   VOLUME SEQUENCE DIFFERS
                                         CODE 4           0   -   FILE SEQUENCE DIFFERS
                                         CODE 5           3   -   NO CATALOG ENTRY FOUND
                                         CODE 6           0   -   NO TMC ENTRY FOUND
                                         CODE 7           0   -   AFTER LOCATE, NOT TAPE DEVICE
                                         TOTALS           4
```

Example 2

```
 THURSDAY, FEBRUARY 24, 2005.060                  Y O U R   C O M P A N Y   N A M E                             ROxxxxx PAGE=00001
 TMS REPORT-17 TAPE MANAGEMENT CATALOG (TMC) AND OS/VS CATALOG DIFFERENCES
           DATA SET NAME (TMC)                VSN   VSQ FSQ   EXPDT             DATA SET NAME (CATALOG)             VSN

                               I=YES,RS=DSN,RC=ERRORS,L=YES
```

```
 THURSDAY, FEBRUARY 24, 2005.060                  Y O U R   C O M P A N Y   N A M E                             PAGE=00002
 TMS REPORT-17 TAPE MANAGEMENT CATALOG (TMC) AND OS/VS CATALOG DIFFERENCES
           DATA SET NAME (TMC)                VSN   VSQ FSQ   EXPDT             DATA SET NAME (CATALOG)             VSN

 GDGTAPE.DSN.G0002V00                        001821 001 001 MMMDD YYYY           ********* NO CATALOG ENTRY FOUND *******
 GDGTAPE.DSN.G0005V00                        001829 001 001 MMMDD YYYY           ********* NO CATALOG ENTRY FOUND *******
 SSDDEV.CA8.LOAD                             000008 001 001 USER/199             ********* NO CATALOG ENTRY FOUND *******
 WITRU01.TEST.MFMV.FILE2                     001832 001 002 CATALOG              ********* NO CATALOG ENTRY FOUND *******
```

```
THURSDAY, FEBRUARY 24, 2005.060                  Y O U R   C O M P A N Y   N A M E
 TMS REPORT-17 TAPE MANAGEMENT CATALOG (TMC) AND OS/VS CATALOG DIFFERENCES
           DATA SET NAME (TMC)                VSN   VSQ FSQ   EXPDT             DATA SET NAME (CATALOG)             VSN

                                         ERROR COUNT TOTALS -   DESCRIPTION
                                         CODE 1           0   -   DATA SET NAME DIFFERS (VSN SEQ ONLY)
                                         CODE 2           1   -   VOLUME SERIAL NUMBER DIFFERS (DSN SEQ ONLY)
                                         CODE 3           0   -   VOLUME SEQUENCE DIFFERS
                                         CODE 4           0   -   FILE SEQUENCE DIFFERS
                                         CODE 5           3   -   NO CATALOG ENTRY FOUND
                                         CODE 6           0   -   NO TMC ENTRY FOUND
                                         CODE 7           0   -   AFTER LOCATE, NOT TAPE DEVICE
                                         TOTALS           4
```

Example 3

```
 THURSDAY, FEBRUARY 24, 2005.060                  Y O U R   C O M P A N Y   N A M E                             ROxxxxx PAGE=00001
 TMS REPORT-17 TAPE MANAGEMENT CATALOG (TMC) AND OS/VS CATALOG DIFFERENCES
           DATA SET NAME (TMC)                VSN   VSQ FSQ   EXPDT             DATA SET NAME (CATALOG)             VSN
                               I=YES,L=YES
```

```
THURSDAY, FEBRUARY 24, 2005.060                  Y O U R   C O M P A N Y   N A M E                             PAGE=00002
TMS REPORT-17 TAPE MANAGEMENT CATALOG (TMC) AND OS/VS CATALOG DIFFERENCES
          DATA SET NAME (TMC)                VSN   VSQ FSQ   EXPDT              DATA SET NAME (CATALOG)             VSN

          DATA SET NAME (TMC)                VSN   VSQ FSQ   EXPDT              DATA SET NAME (CATALOG)             VSN   VSQ FSQ ER
GDGTAPE.DSN.G0002V00                        001821 001 0001 MMMDD YYYY           ********* NO CATALOG ENTRY FOUND *******          5
GDGTAPE.DSN.G0003V00                        001825 001 0001 MMMDD YYYY GDGTAPE.DSN.G0003V00                        001825   1   1
GDGTAPE.DSN.G0005V00                        001829 001 0001 MMMDD YYYY           ********* NO CATALOG ENTRY FOUND *******          5
HIJKL01.AIVSGDG.G0013V00                    980015 001 0001 CATALOG    HICFR01.AIVSGDG.G0013V00                    980015   1   1
HIJKL01.NOTAVIS.JOB0130O.FILE1              980024 001 0001 CATALOG    HICFR01.NOTAVIS.JOB0130O.FILE1              980024   1   1
HIJKL01.NOTAVIS.JOB0130O.FILE1              980026 002 0001 CATALOG    HICFR01.NOTAVIS.JOB0130O.FILE1              980026   2   1
HIJKL01.NOTAVIS.JOB0130O.FILE1              980027 003 0001 CATALOG    HICFR01.NOTAVIS.JOB0130O.FILE1              980027   3   1
HIJKL01.NOTAVIS.JOB0130Q.FILE2              980027 001 0002 CATALOG    HICFR01.NOTAVIS.JOB0130Q.FILE2              980027   1   2
HIJKL01.NOTAVIS.JOB0130Q.FILE2              980028 002 0002 CATALOG    HICFR01.NOTAVIS.JOB0130Q.FILE2              980028   2   2
HIJKL01.NOTGDG                              000002 001 0001 CATALOG    HICFR01.NOTGDG                              000002   1   1
SSDDEV.CA8.LOAD                             000008 001 0001 USER/199             ********* NO CATALOG ENTRY FOUND *******          5
WXYZA01.TAPETEST.FILE1                      TDI001 001 0001 MMMDD YYYY WITRU01.TAPETEST.FILE1                      TDI001   1   1
WXYZA01.TAPETEST.FILE1                      TDI002 002 0001 MMMDD YYYY WITRU01.TAPETEST.FILE1                      TDI002   2   1
WXYZA01.TAPETEST.FILE2                      TDI002 001 0002 MMMDD YYYY WITRU01.TAPETEST.FILE2                      TDI002   1   2
WXYZA01.TAPETEST.FILE3                      TDI002 001 0003 MMMDD YYYY WITRU01.TAPETEST.FILE3                      TDI002   1   3
WXYZA01.TAPETEST.FILE4                      TDI002 001 0004 MMMDD YYYY WITRU01.TAPETEST.FILE4                      TDI002   1   4
WXYZA01.TAPETEST.FILE4                      TDI003 002 0004 MMMDD YYYY WITRU01.TAPETEST.FILE4                      TDI003   2   4
WXYZA01.TEST.MFMV.FILE2                     001832 001 0002 CATALOG              ********* NO CATALOG ENTRY FOUND *******          5
```

Example 4

```
THURSDAY, FEBRUARY 24, 2005.060                  Y O U R   C O M P A N Y   N A M E                             ROxxxxx PAGE=00001
TMS REPORT-17 TAPE MANAGEMENT CATALOG (TMC) AND OS/VS CATALOG DIFFERENCES
          DATA SET NAME (TMC)                VSN   VSQ FSQ   EXPDT             DATA SET NAME (CATALOG)             VSN

                                        ERROR COUNT TOTALS -   DESCRIPTION
                                        CODE 1           0   -   DATA SET NAME DIFFERS (VSN SEQ ONLY)
                                        CODE 2           0   -   VOLUME SERIAL NUMBER DIFFERS (DSN SEQ ONLY)
                                        CODE 3           0   -   VOLUME SEQUENCE DIFFERS
                                        CODE 4           0   -   FILE SEQUENCE DIFFERS
                                        CODE 5           0   -   NO CATALOG ENTRY FOUND
                                        CODE 6           0   -   NO TMC ENTRY FOUND
                                        CODE 7           0   -   AFTER LOCATE, NOT TAPE DEVICE
                                        TOTALS           4
```

```
WEDNESDAY, JUNE 22, 2005.171                     Y O U R   C O M P A N Y   N A M E                            PAGE=00002
TMS REPORT-17 TAPE MANAGEMENT CATALOG (TMC) AND OS/VS CATALOG DIFFERENCES
  PARAMETERS USED ON EXECUTE CARD ==> DATEFMT=(MMMDD,YYYY),SYNC,DETAIL

 DSN FOUND ON CATALOG=WXYZA01.TAPETEST.FILE2                       VOLSER=TDI002 FILE SEQUENCE=00002
 DSN FOUND ON CATALOG=WXYZA01.TAPETEST.FILE3                       VOLSER=TDI002 FILE SEQUENCE=00003
 DSN FOUND ON CATALOG=WXYZA01.TAPETEST.FILE4                       VOLSER=TDI002 FILE SEQUENCE=00004
 DSN FOUND ON CATALOG=DEFGH02.TEST.FULL2                           VOLSER=TDI006 FILE SEQUENCE=00002
 DSN FOUND ON CATALOG=DEFGH02.TEST.FULL4                           VOLSER=TDI007 FILE SEQUENCE=00004
 DSN FOUND ON CATALOG=CDEFG01.TMS52.CHKPTDS                        VOLSER=TDI030 FILE SEQUENCE=00001
 DSN -NOT- ON CATALOG=WXYZA01.TESTNL4.FILE1                        VOLSER=TDI040 FILE SEQUENCE=00001
 DSN FOUND ON CATALOG=HIJKL01.NOTAVIS.JOB0130Q.FILE2               VOLSER=980027 FILE SEQUENCE=00002
           8  RECORDS WERE UPDATED
```

TMSOSCAT Report Field Definitions

Control statement image

Input control statement

DATA SET NAME (TMC)

Data set name of tape volume or file

VSN

Volume serial number of tape volume or file

VSQ

Volume sequence number of tape volume

FSQ

File sequence number of file

EXPDT

Expiration date of tape volume or file

DATA SET NAME (CATALOG)

Data set name of tape volume or file

VSN

Volume serial number of tape volume or file

VSQ

Volume sequence number of tape volume

FSQ

File sequence number of file

ERR

Error code indicating discrepancy

ERROR COUNT TOTALS

Number of errors by error code

DESCRIPTION

Description of error codes

TMSPTRS Utility—Maintain Multivolume, Multi-Data Set Chains

Use the TMSPTRS maintenance report program to detect invalid internal CA 1 multivolume and multi-data set chains, or errors that exist in the free chain of unused DSNBs.

Invalid or broken chains may occur as a result of invalid manual updates, abnormal termination of jobs, a system failure during CA 1 processing, 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.

TMSPTRS should be regularly executed to analyze and report on any invalid pointers. Chaining errors left unresolved may result in CA 1 abends of tape jobs or invalid results from CA 1 batch jobs. A completion code of 12 is issued if any errors are detected. A code of zero indicates no errors were found. A case study can be found later in this discussion.

TMSPTRS can also be used to initialize TMC fields which were not available with previous CA 1 versions. It should be run for this purpose immediately after conversion from a lower version. Parameter ONLYTMC or ONLYDSNB must not be specified for this purpose.

The following fields are processed:

- DSNB field DSNBFVSN, pointing to the volume the DSNB file starts on.
- TMC FLAG3 bit TMULTIF, indicating the existence of DSNBs on the volume or in the multivolume chain.

Update control statements for these fields are generated automatically if the appropriate DD statements are found.

Important! These updates are highly sensitive to the structure of multifile/multivolume chains. Do not perform updates until all other pointer errors have been resolved manually and TMSPTRS has been rerun to ensure that correct update statements are generated.

TMSPTRS can be run without issuing a RESERVE against the TMC. A copy of the TMC is made and then updated by applying all changes from the Audit data set that have occurred because the copy operation began. Because it is necessary to apply the Audit records, TMSPTRS should not be run while other CA 1 batch utilities, such as TMSCLEAN and TMSEXPDT, are running.

If the ENQ parameter is specified, the active TMC is RESERVEd for the duration of the copy operation.

CA 1 must be active or batch active to run TMSPTRS.

A S213-10 abend occurs if run with a SYSUT3 allocated with DSNTYPE=LARGE.

In the event TMSPTRS terminates with U0999 abend, R15 will indicate the cause:

- 08 - The Audit data set did not contain Control Record #4 as the first record.
- 12 - A Type 5 Audit record was encountered indicating a structural modification to the TMC was made after it was copied to SYSUT3.
- 16 - A valid Control Record #2 was not found on the TMC.
- 20 - An open error occurred on TMSUPDTE or TMSUDSNB DD.

Note: If a "BLOCKSET Audit" is being used (refer to RO42220 for details on this new formatting option) and TMSPTRS is run without a PARM of ENQ, then on rare occasions you may receive an ERR24, ERR41, or ERR44 that do not actually exist. Running TMSAPEC (recommended) or TMSPTRS with PARM=ENQ eliminates these false errors.

Report Description

TMSPTRS produces TMS Report 29 - Invalid Pointers Report which lists all volume or DSNB records that are in error. A sample of this report appears on n (see page 376). A description of each error and how to correct it is provided later in this section, beginning on Chaining Errors (see page 372).

Job Control Statements

```
//stepname EXEC PGM=TMSPTRS[,PARM='parm']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//SYSPRINT DD  SYSOUT=A
//SYSUT3   DD  UNIT=SYSDA,SPACE=(same as TMC)        *NOTES
//SYSOUT   DD  SYSOUT=A
//SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK06 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//TMSUPDTE DD  DSN=tmsupdte.control.statements,
//             DISP=(NEW,CATLG,DELETE),
//             UNIT=unit,SPACE=(space),DCB=BLKSIZE=n*80    *NOTES
//TMSUDSNB DD  DSN=tmsudsnb.control.statements,
//             DISP=(NEW,CATLG,DELETE),
//             UNIT=unit,SPACE=(space),DCB=BLKSIZE=n*80    *NOTES
```

Notes:

- The BLKSIZE can be any multiple of the LRECL, which is 80. If no BLKSIZE is specified and the release of the operating system supports system determined blocksize, the operating system selects the blocksize. Otherwise the BLKSIZE defaults to 32640 for tape or 8840 for DASD devices.
- The SYSUT3 data set holds a copy of the TMC. It can be blocked, regardless of whether the TMC is blocked or not. The BLKSIZE can be any multiple of the LRECL, which is 340.

Parameter Definitions

If no parameter is specified, all multivolume and multi-data set pointers are analyzed.

ONLYTMC *note

Directs TMSPTRS to analyze only multivolume pointers.

ONLYDSNB *note

Directs TMSPTRS to analyze only multi-data set pointers.

Note: While ONLYTMC and ONLYDSNB will each run faster than without any parameter, the results are not as complete. Some types of errors can only be found when neither parameter is used, which is the recommended way to run TMSPTRS.

ENQ

Reserves the TMC for the length of the copy operation.

Note: If ENQ is used, you must shutdown HSM on all systems that share this TMC.

JCL Considerations

TMSUPDTE DD

Optional DD statement. If present, a sequential data set is created which contains control statements for utility TMSUPDTE, used for initialization or correction of DSN17 and FLAG3 bit TMULTIF (err79 or err83).

TMSUDSNB DD

Optional DD statement. If present, a sequential data set is created which contains control statements for utility TMSUDSNB, used for initialization of field DSNBFVSN (err78).

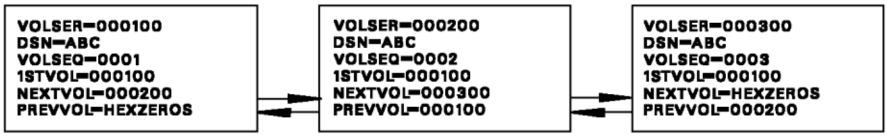
Note: No output is written to either TMSUPDTE or TMSUDSNB if a parameter of ONLYTMC or ONLYDSNB is used.

Control Statement Specification

There are no control statements for TMSPTRS.

TMC Chaining Structures

Understanding the various structures found in the TMC is helpful in correcting any errors reported. Multivolume chains are carried in five fields of each TMC Volume record associated with the data set, VOLSEQ, VOLSER, 1STVOL, NEXTVOL and PREVVOL, using forward and backward pointers. All Volume records in a volume chain must contain the same value in the 1STVOL field.

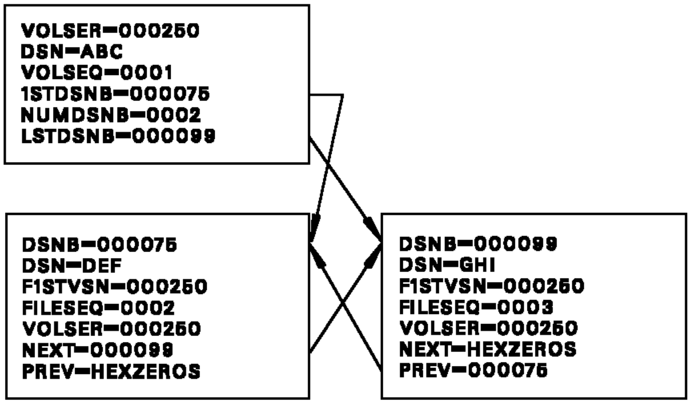


VOLSER=000100
DSN=ABC
VOLSEQ=0001
1STVOL=000100
NEXTVOL=000200
PREVVOL=HEXZEROS

VOLSER=000200
DSN=ABC
VOLSEQ=0002
1STVOL=000100
NEXTVOL=000300
PREVVOL=000100

VOLSER=000300
DSN=ABC
VOLSEQ=0003
1STVOL=000100
NEXTVOL=HEXZEROS
PREVVOL=000200

Multi-data set chains are reflected in three fields of the TMC Volume record, 1STDSNB, LSTDSNB, and NUMDSNB, and in three fields of the DSNB record, VOLSER, PREV and NEXT. DSNB chains are constructed with forward and backward pointers. All DSNB records in a DSNB chain must contain the same value in the VOLSER field.

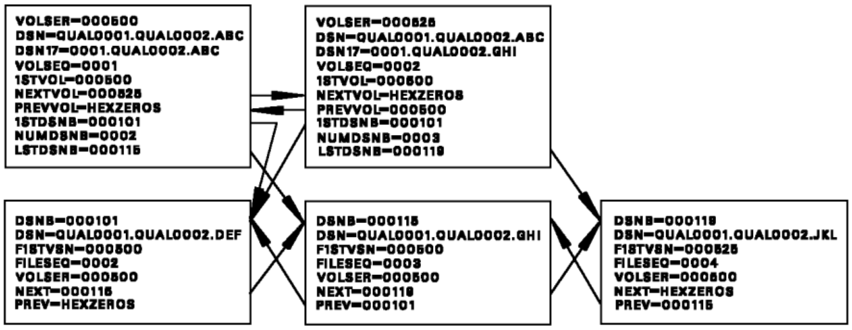


VOLSER=000250
DSN=ABC
VOLSEQ=0001
1STDSNB=000075
NUMDSNB=0002
LSTDSNB=000099

DSNB=000075
DSN=DEF
F1STVSN=000250
FILESEQ=0002
VOLSER=000250
NEXT=000099
PREV=HEXZEROS

DSNB=000099
DSN=GHI
F1STVSN=000250
FILESEQ=0003
VOLSER=000250
NEXT=HEXZEROS
PREV=000075

Multivolume/multi-data set (aggregate) chains use all chaining fields in TMC and DSNB records. DSNB chains are constructed with forward and backward pointers starting with the first volume that has the second file created on it. The DSNB total must be reflected in the last volume associated with the aggregate.



```
VOLSER=000500
DSN=QUAL0001.QUAL0002.ABC
DSN17=0001.QUAL0002.ABC
VOLSEQ=0001
1STVOL=000500
NEXTVOL=000525
PREVVOL=HEXZEROS
1STDSNB=000101
NUMDSNB=0002
LSTDSNB=000115
```

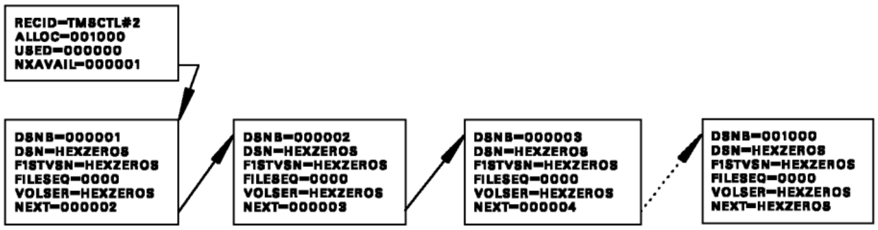
```
VOLSER=000525
DSN=QUAL0001.QUAL0002.ABC
DSN17=0001.QUAL0002.GHI
VOLSEQ=0002
1STVOL=000500
NEXTVOL=HEXZEROS
PREVVOL=000500
1STDSNB=000101
NUMDSNB=0003
LSTDSNB=000119
```

```
DSNB=000101
DSN=QUAL0001.QUAL0002.DEF
F1STVSN=000500
FILESEQ=0002
VOLSER=000500
NEXT=000115
PREV=HEXZEROS
```

```
DSNB=000115
DSN=QUAL0001.QUAL0002.GHI
F1STVSN=000500
FILESEQ=0003
VOLSER=000500
NEXT=000119
PREV=000101
```

```
DSNB=000119
DSN=QUAL0001.QUAL0002.JKL
F1STVSN=000525
FILESEQ=0004
VOLSER=000500
NEXT=HEXZEROS
PREV=000115
```

The free chain of DSNBs, that is, those DSNBs that are unused and available to track new secondary data sets, is maintained through a pointer field, NXAVAIL, contained in Control Record #2 (TMSCTL#2) of the TMC. In a newly formatted TMC, the DSNBs are consecutive off of Control Record #2. All unused DSNB records in the free chain should have bit DSNBACTV of the active indicator field (ACTIND) turned off. The last DSNB should never be used and must always contain HEXZEROS in the NEXT field.



```
RECID=TMSCTL#2
ALLOC=001000
USED=000000
NXAVAIL=000001
```

```
DSNB=000001
DSN=HEXZEROS
F1STVSN=HEXZEROS
FILESEQ=0000
VOLSER=HEXZEROS
NEXT=000002
```

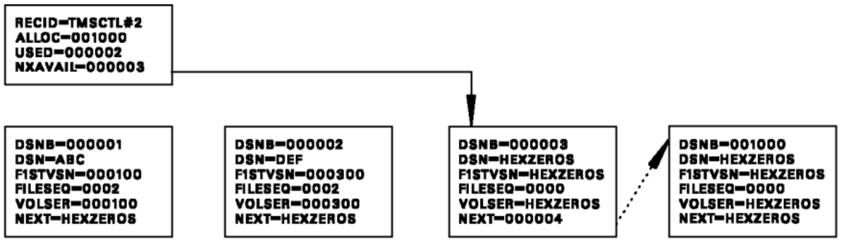
```
DSNB=000002
DSN=HEXZEROS
F1STVSN=HEXZEROS
FILESEQ=0000
VOLSER=HEXZEROS
NEXT=000003
```

```
DSNB=000003
DSN=HEXZEROS
F1STVSN=HEXZEROS
FILESEQ=0000
VOLSER=HEXZEROS
NEXT=000004
```

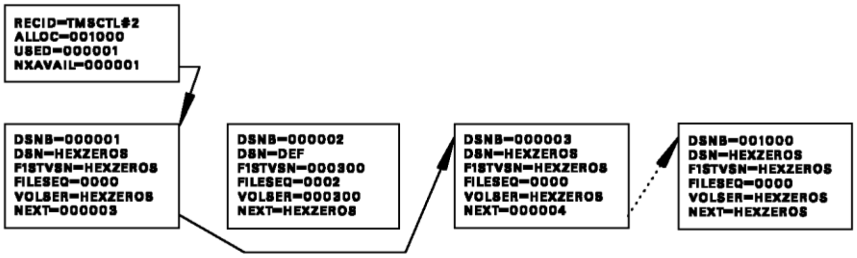
```
DSNB=001000
DSN=HEXZEROS
F1STVSN=HEXZEROS
FILESEQ=0000
VOLSER=HEXZEROS
NEXT=HEXZEROS
```

Note: The PREV field in the DSNB is not used in the free chain.

As DSNBs are allocated, the NXAVAIL field in Control Record #2 is altered to reflect the next unused DSNB in the free chain. The USED field is incremented by one to reflect the number of DSNB records already used. The active indicator (ACTIND) is turned on in the DSNB.



During scratch processing, TMSCLEAN clears the DSNB active indicator (ACTIND) and returns the DSNB to the front of the free chain. The NEXT field in the DSNB record is updated to reflect the current NXAVAIL value in Control Record #2, the NXAVAIL field is updated with the DSNB number being returned to the free chain, and the USED field is decremented by one.



Note: After the DSNBs have been used and returned to the free chain, the DSNBs are no longer consecutive off of Control Record #2.

Chaining Errors

TMSPTRS provides information on any errors with the chaining structures previously described. In most cases, CA 1 online access facilities (CA 1 ISPF or TIQ) of CA 1 allow corrections to be made to the database or three batch utilities can be used:

1. TMSUPDTE updates erroneous fields in Volume records.
2. TMSUDSNB updates erroneous fields in DSNB records or Control Record #2 (TMSCTL#2).
3. TMSAGGR clears and returns invalid used DSNBs to the free chain and constructs new DSNB chains for Volume records.

Additionally, the TMSBINQ utility can be used to provide detailed reporting of erroneous records when online inquiries are not practical.

If TMSPTRS ends with a nonzero condition code, errors are present in the TMC. The four types of chaining errors are:

1. Multivolume Chaining Errors

   Research of errors in multivolume chains involves determining whether:

   a. The volume(s) contains live data and should be part of a multivolume group.

   b. The volume(s) no longer has data which needs to be retained and should be returned to the scratch pool.

   This information can be obtained by displaying Volume record(s) in the TMC, examining system catalog information, dumping the contents of the tape, or a combination of the three.

2. Multi-Data Set Chaining Errors

   Research of errors in multi-data set chains involves determining whether:

   a. The volume(s) contains live secondary data sets and should have a proper chain of DSNBs.

   b. The volume(s) does not have secondary data sets and should be returned to scratch status, or the DSNBs referencing the volume should be cleared and returned to the free chain.

   This information can be obtained by displaying volume and DSNB record(s) in the TMC, examining system catalog information, dumping the contents of the tape, or a combination of the three.

3. Aggregate Chaining Errors

4. Control Record #2/DSNB Free Chain Errors

Correction Control Statement Examples

TMSUPDTE Control Statements - Correcting Multivolume Pointers

The samples below assume that PARM=NODSN was coded.

- Clear a volume with invalid multivolume pointers to make available for scratch processing:

```
VOL nnnnnn,NOCHAIN
REP DSN=HEXZEROS
REP 1STVOL=HEXZEROS
REP NEXTVOL=HEXZEROS
REP PREVVOL=HEXZEROS
REP VOLSEQ=1
REP EXPDT=current-date
```

- Repair a broken multivolume chain:

```
VOL nnnnnn,NOCHAIN
REP DSN=valid-data-set-name
REP 1STVOL=valid-VOLSER
REP NEXTVOL=valid-VOLSER
REP PREVVOL=valid-VOLSER
REP VOLSEQ=valid-volseq
```

TMSUPDTE Control Statements - Correcting Multi-Data Set Pointers

The samples below assume that PARM=NODSN was coded.

- Clear references to invalid DSNBs:

```
VOL nnnnnn,NOCHAIN
REP 1STDSNB=HEXZEROS
REP NUMDSNB=HEXZEROS
```

- Repair invalid DSNB pointers:

```
VOL nnnnnn,NOCHAIN
REP 1STDSNB=valid-DSNB-number
REP NUMDSNB=valid-number-DSNBs
```

- Correct ERR62 errors:

```
VOL nnnnnn-mmmmmm,NOCHAIN
VER 1STDSNB=HEXZEROS
REP LSTDSNB=HEXZEROS
```

Note: nnnnnn is your low VOLSER in the TMC and mmmmmm is your high VOLSER. You may have more updates than errors as any volume having a 1STDSNB value of HEXZEROS will update the LSTDSNB field. It is recommended that TMSCOPY be run before executing this procedure.

TMSAGGR Control Statements - DSNB Maintenance

- Clear and return an invalid used DSNB to the free chain:

  ```
  DEL nnnnnn
  ```

- Purge all DSNBs from a volume:

  ```
  VOL nnnnnn,PURGE
  ```

- Attach a DSNB chain to a volume:

  ```
  VOL nnnnnn
  DSN=file.two.dsn,EXPDT=yyyy/ddd,...
  DSN=file.three.dsn,EXPDT=yyyy/ddd,...
  DSN=file.four.dsn,EXPDT=yyyy/ddd,...
  DSN=file.five.dsn,EXPDT=yyyy/ddd,...
  ```

TMSUDSNB Control Statements - DSNB Maintenance

The samples below assume PARM=NOVER was coded.

- Correct fields in a DSNB record:

  ```
  PASSWORD=system-password
  DSNB nnnnnn
  REP DSN=valid-data-set-name
  REP NEXT=next-valid-DSNB
  REP FILSEQ=valid-file-sequence
  REP VOLSER=valid-volume-serial
  REP PREV=valid-previous-DSNB
  ```

- Clear fields in a DSNB record:

  ```
  PASSWORD=system-password
  DSNB nnnnnn
  REP NEXT=HEXZEROS
  REP FILSEQ=HEXZEROS
  REP VOLSER=HEXZEROS
  REP DSN=HEXZEROS
  REP PREV=HEXZEROS
  ```

TMSUDSNB Control Statements - Control Record #2 Maintenance

Updates to TMSCTL#2 should be performed only when there is no TMC activity. It is recommended that the verify operation be performed when updating TMSCTL#2. The samples below assume that PARM=NOVER was not coded.

- Update the NXAVAIL pointer to an unused DSNB chain:

```
PASSWORD=system-password
CTL2
VER RECID=TMSCTL#2
VER NXAVAIL=current-value
REP NXAVAIL=correct-DSNB-number
```

- Update the USED field to the correct value:

```
PASSWORD=system-password
CTL2
VER RECID=TMSCTL#2
VER USED=current-value
REP USED=correct-number-in-use
```

TMSUDSNB Control Statements - DSNB Free Chain Maintenance

Updates to TMSCTL#2 should be performed only when there is no TMC activity. It is recommended that the verify operation be performed when updating TMSCTL#2. The samples below assume that PARM=NOVER was not coded.

- Add unused DSNB(s) not currently in the free chain:

```
PASSWORD=system-password
CTL2
VER RECID=TMSCTL#2
VER NXAVAIL=current-NXAVAIL-value
REP NXAVAIL=unused-DSNB-number
DSNB unused-DSNB-number
VER NEXT=current-NEXT-value
REP NEXT=current-NXAVAIL-value
```

- Append an unused DSNB chain that chains through end-of-chain to the last DSNB referenced by Control Record #2:

```
PASSWORD=system-password
DSNB last-DSNB-from-CTL2
VER NEXT=value-in-unused-DSNB
REP NEXT=first-DSNB-in-unused-chain
```

TMS Report 29 - Invalid Pointers

```
SUNDAY, JUNE 26, 2005.175                        Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT 29 - INVALID POINTERS     OPTIONS: TMC DSNB, NO ENQ

ERR39  VOLUME 501908 HAS VOLUME SEQUENCE OF 003 AND VOLUME IS NOT CHAINED
ERR39  VOLUME 501933 HAS VOLUME SEQUENCE OF 002 AND VOLUME IS NOT CHAINED
ERR39  VOLUME 501983 HAS VOLUME SEQUENCE OF 003 AND VOLUME IS NOT CHAINED
ERR39  VOLUME 502006 HAS VOLUME SEQUENCE OF 003 AND VOLUME IS NOT CHAINED
ERR41  015020 DSNBS USED(HDR), 019017 DSNBS ACTUALLY USED
ERR24  UNUSED DSNB 000089 IS NOT IN CHAIN AND CHAINS THRU DSNB 000087, 000095 DSNBS IN CHAIN
ERR24  UNUSED DSNB 000105 IS NOT IN CHAIN AND CHAINS THRU DSNB 001023, 000919 DSNBS IN CHAIN
ERR24  UNUSED DSNB 001024 IS NOT IN CHAIN AND CHAINS THRU DSNB 001416, 000393 DSNBS IN CHAIN
ERR24  UNUSED DSNB 021418 IS NOT IN CHAIN AND CHAINS THRU END-OF-CHAIN, 003582 DSNBS IN CHAIN
ERR24  UNUSED DSNB 001502 IS NOT IN CHAIN AND CHAINS THRU DSNB 001502, 000001 DSNBS IN CHAIN
ERR31  VOL 000087 POINTS TO DSNB 000000 NOT TO 001417
ERR31  VOL 000102 POINTS TO DSNB 000000 NOT TO 000088
ERR44  TMSCTL#2 POINTS TO USED DSNB 001023

*******************************************************************************************************************************


         END OF REPORT 29  -  TOTAL ITEMS LISTED=       17
```

TMSPTRS Report Field Definitions

TMS REPORT 29 will list out the options specified for this run of TMSPTRS followed by the specific error messages for any pointer errors found. There are no predefined fields on TMS REPORT 29.

TMSPULL Utility—Produce Library Pull and Setup Lists

Use the TMSPULL utility to prestage (set up) tape librarian jobs ahead of time.

The CA 1 prestaging facility is designed to aid in this task. A partitioned data set (PDS) can be created which contains the required information to selectively pull tapes by job or by shift.

Information for each job must be furnished to this program through a PDS with the job name as the member name. Each member contains the data sets to be pulled for that job, the shift it normally runs on, and the cart on which the tapes should be placed.

CA 1 must be active or batch active to run TMSPULL. TMSPULL can be run with CA 1 inactive only if you are using the system catalog for input (PARM=OSCATLG).

Report Description

The TMSPULL program generates two reports: TMS Report 37 - Pull List and TMS Report 38 - Setup List. Samples of these reports appear on TMS Report 37 - Pull List (see page 381).

The Pull List is a list of volume serial numbers for the tape librarian to pull from the library in shift, cart and volume serial number sequence. Other fields on the report include volume sequence number, job name, data set name, vault name and slot number. Slots are automatically assigned by cart. The Setup List is in shift, job name, and volume serial number sequence. Other fields reported on include data set name, volume sequence number, cart and slot number. This list informs the operator which cart and slots contain tapes.

Slot/Cart Record and Data Set Name Record

The slot/cart record indicates the shift during which the job will run and the cart on which the tape should be placed. There should be only one slot/cart record per job.

The data set name record contains the data set name, creating job name, and volume serial information. There should be a data set name record for each input tape data set. Scratch tapes for output data sets may also be pulled.

Position 72 of each record must be blank. Positions 73 through 80 may be used for sequence numbers. Continuation statements are not allowed. A comment statement is indicated by an asterisk (*) in position 1.

Slot/Cart Record Format

The format for the slot/cart record is:

```
1      7
```

```
[SHIFT=n][,CART=xxxxxxxx]
```

SHIFT

Indicates the shift during which the job should run. If SHIFT is specified, it must be the first keyword on the first record in the PDS job member. The value of n is one character, 0 through 9 or A through Z.

CART

Indicates the name or number of the cart which is to contain the pulled tapes. The value of xxxxxxxx is one to eight characters, A through Z and 0 through 9. If CART is specified, this program assigns slot numbers on that cart.

Data Set Name Record Format

The format for the data set name record is:

```
1   5

                                          {SCRTCH }
DSN=dsname[(-ccc)][,CJOB=jobname][,VOLSER= {vvvvvv }]
```

DSN

Specifies the data set name of the tape to be pulled. For simple data sets, only the most recent creation is pulled. If the data set name contains special characters or blanks, it must be enclosed within delimiters that have a character value less than A.

ccc

Is a numeric value of 0 through 255. The current copy (0) is the default. If using the system catalog, the relative generation can be specified for GDGs, such as (-1). If the data set name contains special characters or blanks, it must be enclosed within delimiters that have a character value less than A.

CJOB

Specifies the job name that created the data set.

VOLSER

Specifies volume serial number to pull. SCRTCH is specified to indicate output.

Example:

The following is an example of a PDS member (job B7USELIB):

```
SHIFT=2,CART=A01
DSN=B7USELIB.TRAN(0),CJOB=B6USELIB
DSN=B7USELIB.POST,CJOB=B7USELIB,VOLSER=SCRTCH
DSN=USELIB.MSTR,CJOB=CUSTMR
DSN=*$REPORT*,VOLSER=SCRTCH
```

Job Control Statements

```
//stepname EXEC PGM=TMSPULL,
//              PARM='parm'
//STEPLIB  DD   DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD   SYSOUT=A
//TMSPULL  DD   SYSOUT=A
//TMSETUP  DD   SYSOUT=A
//SYSLIB   DD   DSN=pull.list.pds,DISP=SHR
//SYSUT1   DD   UNIT=SYSDA,
//              SPACE=(TRK,(20,10))
//SYSOUT   DD   SYSOUT=A
//SYSPRINT DD   SYSOUT=A
//SORTLIB  DD   DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(10,5),,CONTIG)
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(10,5),,CONTIG)
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(10,5),,CONTIG)
//SORTWK04 DD   UNIT=SYSDA,SPACE=(CYL,(10,5),,CONTIG)
//SORTWK05 DD   UNIT=SYSDA,SPACE=(CYL,(10,5),,CONTIG)
//SORTWK06 DD   UNIT=SYSDA,SPACE=(CYL,(10,5),,CONTIG)
//SYSIN    DD   *
(control statements here)
/*
```

Parameter Definitions

Using the parameter information specified in the JCL, TMSPULL determines from the system catalog or the TMC which tape volumes to pull. Only one parameter may be specified.

OSCATLG

Search only the system catalog for data set volumes.

TMCATLG

Search only the TMC for data set volumes.

OTCATLG

Search both the TMC and system catalog for data set volumes. If both catalogs are specified, the system catalog is checked first. The TMC is then checked for any data sets not located on the system catalog.

JCL Considerations

SYSLIB DD

The SYSLIB PDS is for TMSPULL and should contain only TMSPULL job and shift information. Do not specify a PDS member name. The entire data set must be read by TMSPULL.

Control Statement Specification

The tape librarian can request the selection of volumes by job or by shift. Selection cannot be for both job and shift in the same step. Selection by shift pulls all jobs that have specified that shift. The absence of a SYSIN DD denotes the pulling of all jobs in the PDS.

INCLUDE and EXCLUDE statements cannot be included in the same SYSIN data set or an error message is issued. The maximum is 4094 INCLUDE or EXCLUDE statements.

INCLUDE and EXCLUDE Control Statement Format

The formats for the INCLUDE and EXCLUDE control statements are:

```
1

INCLUDE SHIFT=n
EXCLUDE JOB=jobname
```

Control Statement Definitions

INCLUDE SHIFT

Defines whether a shift is to be included in the Pull List. INCLUDE must begin in position 1 and must be followed by a space. The value of n indicates the shift and is one character. Valid characters are A through Z or 0 through 9.

EXCLUDE JOB

Defines whether a job is to be excluded from the Pull List. EXCLUDE must begin in position 1 and must be followed by a space. The jobname identifies the job name of the job to be excluded and may be from one to eight characters. Valid characters are A through Z, 0 through 9, and special characters @, # and $.

Control Statement Examples

The Pull List PDS contains job members for shifts 1, 2 and 3. The Pull List is to be generated for shifts 2 and 3.

```
INCLUDE SHIFT=2
INCLUDE SHIFT=3
```

The Pull List PDS contains jobs for only one shift. The Pull List is to be generated for all jobs except B7USELIB and B7ACCLST.

```
EXCLUDE JOB=B7USELIB
EXCLUDE JOB=B7ACCLST
```

TMS Report 37 - Pull List

```
WEDNESDAY, JULY 27, 2005.207                 Y O U R   C O M P A N Y   N A M E                                 PAGE=00003

TMS REPORT 37 - PULL LIST

VOLSER    CART   SLOT SHIFT JOB-NAME  VOLSEQ   DATA SET NAME                              VAULT  SLOT

100001  BLUE      001   2   ADMINPWY   001   ADMIN.PAYR.DLY.G0406V00                       SAFE      1
100002  BLUE      002   2   ADMINPMY   001   ADMIN.PAYR.WLY.G0088V00                       SAFE      2
100003  BLUE      003   2   ADMINPWY   001   ADMIN.PAYR.DLY.G0405V00                       SAFE      3
100004  BLUE      004   2   ADMINPMY   001   ADMIN.PAYR.WLY.G0087V00                       SAFE      4
100005  BLUE      005   2   ADMINPWY   001   ADMIN.PAYR.DLY.G0404V00                       SAFE      5
100009  BLUE      006   2   ADMINPWY   001   ADMIN.PAYR.DLY.G0402V00                       SAFE      7
100010  BLUE      007   2   ADMINPMY   001   ADMIN.PAYR.WLY.G0086V00                       SAFE      8
100012  BLUE      008   2   ADMINPMY   001   ADMIN.PAYR.WLY.G0085V00                       SAFE     10
100017  BLUE      009   2   ADMINPWY   001   ADMIN.PAYR.DLY.G0403V00                       SAFE     13
100021  BLUE      010   2   ADMINPWY   001   ADMIN.PAYR.DLY.G0401V00                       SAFE     15
100033  BLUE      011   2   ADMINPWY   001   ADMIN.PAYR.DLY.G0400V00                       SAFE     19
100045  BLUE      012   2   ADMINPQY   001   ADMIN.PAYR.MLY.G0044V00                       SAFE     21
100046  BLUE      013   2   ADMINPQY   001   ADMIN.PAYR.MLY.G0043V00                       SAFE     27
100050  BLUE      014   2   ADMINPQY   001   ADMIN.PAYR.MLY.G0045V00                       SAFE     22
```

TMS Report 38 - Setup List

```
TMS REPORT 38 - SETUP LIST

JOB NAME           DATA SET NAME                      VOLSER VOLSEQ  CART   SLOT  SHIFT

ADMINPMY  ADMIN.PAYR.WLY.G0088V00                     100002  001  BLUE      002   2
ADMINPMY  ADMIN.PAYR.WLY.G0087V00                     100004  001  BLUE      004   2
ADMINPMY  ADMIN.PAYR.WLY.G0086V00                     100010  001  BLUE      007   2
ADMINPMY  ADMIN.PAYR.WLY.G0085V00                     100012  001  BLUE      008   2
ADMINPQY  ADMIN.PAYR.MLY.G0044V00                     100045  001  BLUE      012   2
ADMINPQY  ADMIN.PAYR.MLY.G0043V00                     100046  001  BLUE      013   2
ADMINPQY  ADMIN.PAYR.MLY.G0045V00                     100050  001  BLUE      014   2
ADMINPWY  ADMIN.PAYR.DLY.G0406V00                     100001  001  BLUE      001   2
ADMINPWY  ADMIN.PAYR.DLY.G0405V00                     100003  001  BLUE      003   2
ADMINPWY  ADMIN.PAYR.DLY.G0404V00                     100005  001  BLUE      005   2
ADMINPWY  ADMIN.PAYR.DLY.G0402V00                     100009  001  BLUE      006   2
ADMINPWY  ADMIN.PAYR.DLY.G0403V00                     100017  001  BLUE      009   2
ADMINPWY  ADMIN.PAYR.DLY.G0401V00                     100021  001  BLUE      010   2
ADMINPWY  ADMIN.PAYR.DLY.G0400V00                     100033  001  BLUE      011   2
```

TMSPULL Report Field Definitions

VOLSER

Volume serial number of tape volume

CART

Cart number for tape volume

SLOT

Slot number for tape volume

JOB-NAME

Job name for tape volume

VOLSEQ

Volume sequence number of tape volume

DATA SET NAME

Data set name of tape volume

VAULT

Out-of-area code of tape volume

SLOT

Slot number of tape volume

SHIFT

Shift identifier for tape volume

TMSREMOV Utility—Remove TMC Volume Records

The functionality provided by TMSREMOV is included in the TMSXTEND utility. TMSXTEND provides a simpler way to remove a range of volumes. Refer to the discussion of the benefits and restrictions associated with TMSXTEND later in this manual. After creating a new TMC using the TMSXTEND utility, you must use that utility to add or remove any volume ranges or DSNBs.

TMSREMOV is used to physically remove a range of Volume records from the TMC. Step-by-step instructions for this procedure are detailed on TMSREMOV Procedure (see page 383). The volumes must be in an inactive (DELETE) status, which can be established by TMSDELET once the volumes have been scratched by TMSCLEAN. The flowchart on Control Statement Specification (see page 385) provides an overview of this operation.

CA 1 must be batch active to run TMSREMOV.

Important! This program should be used only by the personnel responsible for the installation and maintenance of CA 1.

Report Description

The volumes removed from the TMC are listed on TMS Report 39 - Purged TMC Records. TMS Report 40 - Purge Errors details any errors discovered during processing. Samples of these reports appear on TMS Report 39 - Purged TMC Records (see page 386).

TMSREMOV Procedure

To remove a range of volume records from the TMC, use the TMSREMOV utility.

Follow these steps:

1. Stop tape processing on all CPUs. If the volume has already been through OPEN processing and sufficient tape remains on the reel, you can leave online transaction tapes (CICS, IMS, and so on) on the drives. The condition prevents the invocation of end-of-volume processing during the procedure (all eight steps).
2. Execute TMSCOPY to provide a backup of the TMC and Audit data sets.
3. Allocate a target TMC on a device other than the production TMC using a target data set name and the following DCB parameters:

   ```
   DCB=(RECFM=FB,LRECL=340,BLKSIZE=n*340,DSORG=PSU)
   ```

   Note: The BLKSIZE can be any multiple of the LRECL, 340, but not 680. When blocking the TMC, the smallest valid block size is 1020. When formatting a small blocked test TMC, ensure that it consists of at least two blocks. A recommended value is BLKSIZE=8840. For small block sizes, you can add an appropriate BUFNO=nn value to the DCB parameter to improve performance in batch utilities.
4. Format the target TMC with TMSFORMT(PARM='NEW'), specifying only the volume ranges you wish to retain in production. The number of DSNBs that you allocate to the target TMC must match the number of DSNBs that is allocated to the production TMC. Determine the current DSNB allocation using either the online facilities or TMSBINQ (executed with the HEADERS control statement). The current DSNB allocation is reflected in TMC control record #2.
5. Request the DBS subtask to RELEASE the TMC through the CTS address space commands.
6. Rename the production TMC to a backup name and rename the target TMC to the production name. Catalog both the backup TMC and the renamed production TMC on all CPUs sharing the TMC. The renamed production TMC is used as input to the TMSREMOV process.
7. Execute TMSINIT on all CPUs sharing the TMC, and place CA 1 in batch active mode. CA 1 resident module addresses the target TMC as the production database once reinitialized. TMSREMOV uses this empty database to receive the input from the renamed production TMC. TMSREMOV copies the records that are not reflected (formatted) in the target TMC to a backup data set. The utility also copies the records that are read from the production TMC to a backup data set.

8. Execute TMSREMOV. Check for errors on TMS Report 40. If errors exist, the target TMC can be recataloged back to the original work name. The renamed production TMC can be recataloged to the production name. To reestablish the production TMC in the resident modules, execute TMSINIT on all CPUs. Tape processing can resume while the errors are corrected, then these steps can be repeated. If no errors are present, submit the TMSEARL CTAPJCL member using the TMEVSNM CTAPECPB member as input to the SYSIN DD. You generate an inventory of the new TMC. If the TMSREMOV execution encounters errors, CA 1 issues an S1yy-104 abend. The target TMC is rendered unusable in a production environment.

   Note: You cannot remove DSNB records with TMSREMOV. To remove DSNB records, use the CA 1 split and merge facilities or TMSXTEND.

9. Reinitialize CA 1 into an active state on all CPUs with TMSINIT.

10. Request the DBS subtask to OBTAIN the TMC through the CTS address space commands.

11. Back up the new TMC using TMSCOPY and resume tape processing.

More information:

TMSINIT Utility—Activate, Deactivate CA 1 (see page 328)

Job Control Statements

```
//stepname EXEC PGM=TMSREMOV[,PARM='DATEFMT=(fmt)']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//OLDTMC   DD  DSN=OLD.TMC,DISP=SHR
//TMC      DD  DSN=NEW.TMC,DISP=SHR
//PURGDLST DD  SYSOUT=A
//PURGDVOL DD  DSN=CAI.CA1.PURGED.VOLUMES,
//             [DCB=BLKSIZE=n*340,]                         *NOTE
//             UNIT=TAPE,DISP=(NEW,KEEP)
```

Note: The BLKSIZE may be any multiple of the LRECL, which is 340. If no BLKSIZE is specified and the release of the operating system supports system determined blocksize, the operating system will select the blocksize. Otherwise the BLKSIZE for PURGDVOL defaults to 32640 for tape or 8840 for DASD devices.

Parameter Definitions

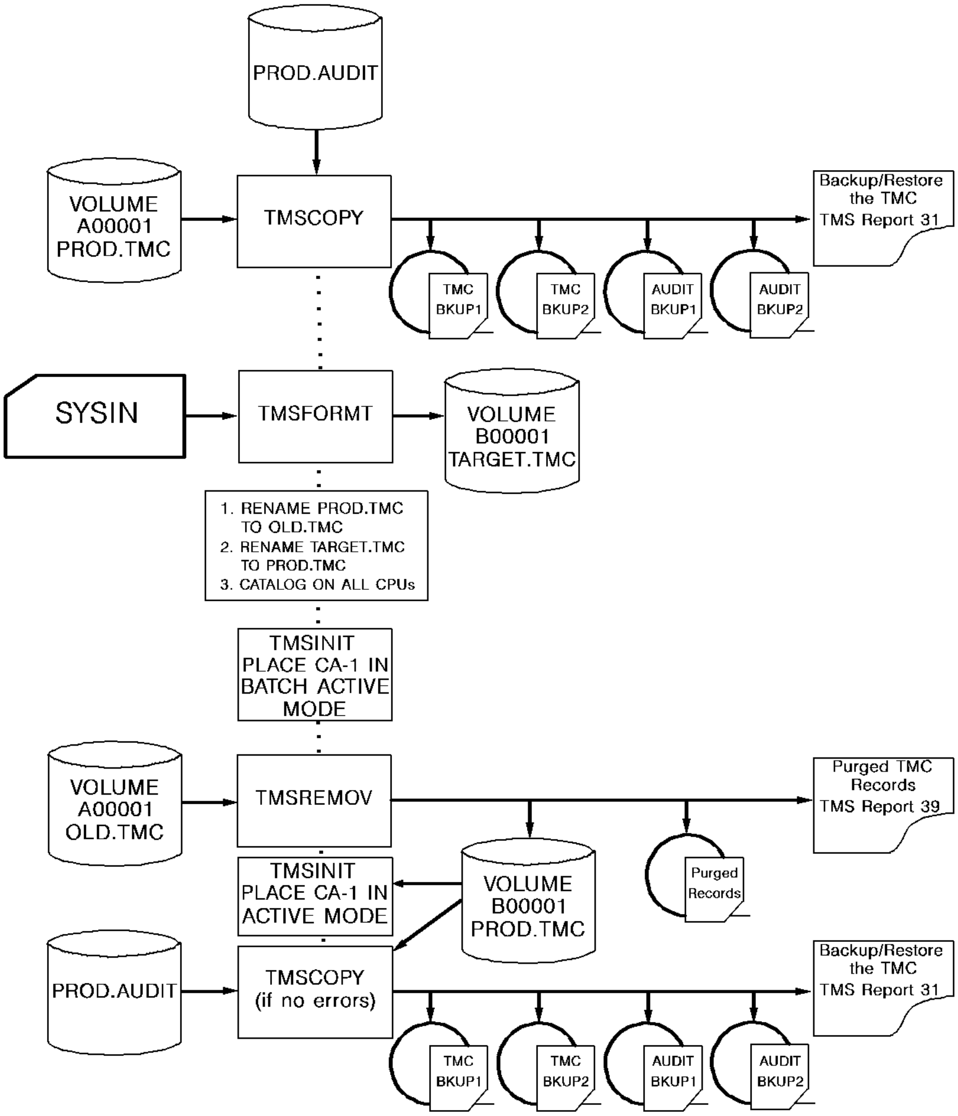
DATEFMT=(fmt)

Specifies the date format to be printed on TMS Reports 39 and 40. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

Control Statement Specification

There are no control statements for TMSREMOV.

The following diagram illustrates Removing Volume Records from the TMC:



TMS Report 39 - Purged TMC Records

```
WEDNESDAY, JULY 27, 2005.207                    Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT 39 - PURGED TMC RECORDS

103000  103001  103002  103003  103004  103005  103006  103007  103008  103009  103010  103011  103012  103013  103014  103015
103016  103017  103018  103019  103020  103021  103022  103023  103024  103025  103026  103027  103028  103029  103030  103031
103032  103033  103034  103035  103036  103037  103038  103039  103040  103041  103042  103043  103044  103045  103046  103047
103048  103049  103050  103051  103052  103053  103054  103055  103056  103057  103058  103059  103060  103061  103062  103063
103064  103065  103066  103067  103068  103069  103070  103071  103072  103073  103074  103075  103076  103077  103078  103079
103080  103081  103082  103083  103084  103085  103086  103087  103088  103089  103090  103091  103092  103093  103094  103095
```

TMS Report 40 - Purge Errors

Any errors encountered are listed on the report with the totals.

```
WEDNESDAY, JULY 27, 2005.207                    Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

TMS REPORT 40 - PURGE ERRORS

TMC RECORDS WRITTEN        7,000     TMC RECORDS DELETED        5,000
DSNBS WRITTEN             25,000     *** ERROR COUNT ***            0
```

TMSREMOV Report Field Definitions

TMC RECORDS WRITTEN

Number of Volume records written to new TMC

DSNBS WRITTEN

Number of DSNB records written to new TMC

TMC RECORDS DELETED

Number of Volume records written to purge data set

***** ERROR COUNT *****

Number of processing errors

TMSRINIT - TMC EXTEND MONITOR

The TMSRINIT program is part of the TMC Extend feature. The CA 1 SVC TMC Extend feature detects when an extend function has been requested and to invoke the routines that will be used to complete the switch-over to the new TMC. TMSRINIT provides functionality similar to the SVC, detecting that a switch-over has been requested and it then invokes the routines to complete the switch-over. You will be prompted to run the TMSRINIT during the TMSXTEND procedure. It is recommended that TMSRINIT be set up as a started task.

TMSRINIT may also be run with PARM='STATUS' to provide diagnostic information about the TMC Extend control blocks.

Report Description

TMSRINIT does not create any reports.

Job Control Statements

```
//TMSRINIT  EXEC PGM=TMSRINIT,REGION=0M,TIME=1440,
//          PARM=''
//*         PARM='STATUS       '
//*         PARM='WAIT=00001500'
//CAISNP    DD SYSOUT=*
//SYSUDUMP  DD SYSOUT=*
//
```

Parameter Definitions

PARM=''

Specifies the default monitor mode. The CA 1 Audit control record #4 is examined at 1-minute intervals to determine if a TMC switch is in progress.

PARM='WAIT=HHMMSSTH'

This parameter overrides the wait time between checks of the Audit control record to the value specified.

PARM='STATUS'

Requests that the control blocks created by the TMSXTEND task be dumped in SNAP dumps.

JCL Considerations

CAISNP

Defines the output DD for snap dumps that are requested when PARM='STATUS' is specified.

Completion Codes

RC=0

Normal completion.

RC=4

TMSXTEND was not found running. Verify that TMSXTEND is running and available to update CPUs.

RC=8

Message TMSRI06I is issued. Verify that TMSRINIT has been link-edited as AC=1 and ensure that the load library or the libraries in the steplib are APF Authorized.

Control Statement Specifications

Control

This program does not use any control statements.

Report Field Definitions

TMSRINIT does not produce any reports.

TMSSMF Utility—Recover Tape Activity from SMF Data

Execute the SMF Recovery Utility TMSSMF to capture tape information from SMF data. You can use this data to update the TMC. You recover tracking data that are lost during periods of time when CA 1 was not active. SMF must be active and recording Type 14 (data set input) and Type 15 (data set output) records for the time period during which CA 1 is inactive.

TMSSMF is designed to process a sequential file that the IBM IFASMFDP SMF dump utility or the IBM IFASMFDL creates.

TMSSMF creates update transactions for the TMSUPDTE and TMSAGGR utilities. The information from the type 14 and 15 records is reformatted as control statements for these utilities. Date and time selection parameters are provided to allow you to identify the time period when CA 1 was not active.

CA 1 can be active or batch active when you run TMSSMF.

Report Description

TMSSMF creates TMS REPORT-66 SMF RECOVERY UTILITY OUTPUT to document the actions taken in analyzing and capturing SMF data for CA 1 recovery purposes. Both DETAIL and SUMMARY modes are supported. In SUMMARY mode (which is the default), TMSSMF writes only the total number of records that you select. In DETAIL mode, a report line is written documenting every volume or secondary file (DSNB) to be recovered. The utility prints only the volumes or secondary files that fall within the specified dates and times.

Operational Considerations

Consider the following points when using TMSSMF:

- Back up the TMC before and after running the jobs to process the transactions that TMSSMF creates.
- Stop all tape processing when running the TMSUPDTE and TMSAGGR utilities. At a minimum, do not run the CA 1 daily utilities, TMSPTRS, or the TMSAPEC subtask of the CTS address space. If tape jobs continue running during your processing of the TMSUPDTE and TMSAGGR recovery transactions, any problems are more difficult to investigate.
- The TMSSMF utility may not be able to recover all tape activity during the outage.
- After running the TMSUPDTE and TMSAGGR jobs to process the recovery transactions, start the TMSAPEC (or TMSPTRS) utility to verify the TMC. You can also use a TMSGRW or CA EARL job to print the volumes and files that are created during the outage.
- Tape jobs that ran when CA 1 was not active and that did not have an expiration date use the value that is specified in the TMOOPTxx RP system option.
- If TMSUPDTE ends with a CC=12 and the following message is issued in TMS REPORT-14 *** vvvvvv IS NOT SCRATCH AND SCRATCH KEYWORD IS SPECIFIED, investigate the data on the volser. The volser could have been overwritten while TMS was not active.
- If tape jobs using DISP=MOD processing are run while CA 1 is not active, they cause an End of Volume condition. To address chaining issues, manually update the secondary volumes. The TMSSMF utility cannot create secondary volume chains from the records that SMF creates for DISP=MOD processing.
- Work files that are created during the times specified in TMSSMF are removed from scratch status. An expiration date of the current date is applied (unless TMSEXPDT processing overrides them). With the next execution of TMSCLEAN, the tape is scratched.

Job Control Statements

```
//stepname  EXEC PGM=TMSSMF[,PARM='[MODE=DETAIL|SUMMARY][,DATEFMT=(fmt)]']
//STEPLIB  DD DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD SYSOUT=A
//SYSOUT   DD SYSOUT=A
//SMFINPUT DD DSN=your.smf.extract.file,DISP=SHR,
//            UNIT=xxxxxxxx,VOL=SER=nnnnnn
//TMSUPDTE DD DSN=CAI.CA1.TMSUPDTE.TRANS,DISP=(NEW,CATLG,DELETE),
//            UNIT=SYSDA,VOL=SER=aaaaaa,SPACE=(TRK,(5,5),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=n*80,DSORG=PS)
//TMSAGGR  DD DSN=CAI.CA1.TMSAGGR.TRANS,DISP=(NEW,CATLG,DELETE),
//            UNIT=SYSDA,VOL=SER=bbbbbb,SPACE=(TRK,(20,20),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=n*80,DSORG=PS)
//SYSIN    DD  *
(control statements here)
/*
```

The UNIT= and VOLSER= parameters on the SMFINPUT DD statement are only needed if the SMF DUMP extract file is not cataloged.

Parameter Definitions

MODE=[DETAIL|SUMMARY]

Specifies the level of detail in creating REPORT-66. When you specify DETAIL, every volume and secondary file that is processed is written to the report. When you specify SUMMARY, only summary counters are written to the report.

Default: SUMMARY

DATEFMT=(fmt)

(Optional) Specifies the date format for reports that TMSSMF generates. The date pattern is limited to 10 bytes and must be enclosed in parentheses.

More information:

Override the Preferred Date Pattern (see page 21)

Control Statement Specifications

The TMSSMF utility supports the following control statements. You can specify each control statement only once.

START=(start date)

Defines the starting day of the selection period. The date format should match the pattern specified in the DATEFMT execution parameter. If the DATEFMT pattern is not entered, YYYY/DDD is used. START is combined with the STIME parameter. The statement defines the specific point in time to begin evaluating SMF data tape activity to convert to CA 1 update transactions. If you omit the START control statement, every record up to the END control statement is used (if you specify the END control statement).

STIME=(hh:mm:ss|hh:mm)

Defines the start time of the selection period. The statement is combined with the date specified in the START parameter. The statement defines the point in time before which no SMF data is evaluated for conversion to CA 1 update transactions.

Default: 00:00:00

END=(end date)

Defines the ending day of the selection period. The date format should match the pattern specified in the DATEFMT execution parameter. If the DATEFMT pattern is not entered, YYYY/DDD is used. END is combined with the ETIME parameter. The statement defines the point in time after which no SMF data is evaluated for conversion to CA 1 update transactions. If you omit the END control statement, every record after the START control statement is used (if you specify the START control statement).

ETIME=(hh:mm:ss|hh:mm)

Defines the end time of the selection period. The statement is combined with the date that you specify in the END parameter. The statement defines the point in time after which no SMF data is evaluated for conversion to CA 1 update transactions.

Default: 23:59:59

ACCT=

Defines the ACCT (User Data) field of all tape records that are written to the TMSUPDTE DD. You can better identify tapes that are recovered using the TMSSMF utility. The field is 50 characters maximum and enclosed by single quotes. For volumes that TMSSMF recovers, do not use the User Data field in the TMC volume record for storing job accounting or other information. This information is not available.

Default: blank

Limits: 50 characters

Control Statement Examples

The following example shows the TMSSMF control statements that select SMF records from the period 10:00AM to 7:00PM on August 10, 2012 (Julian date 2012/223). The SMF data set for the entire day is captured on the extract file. These control statements choose only the SMF Type 14 and 15 records for the time that CA 1 was not active (from 10:00AM to 7:00PM).

START=(2012/223)

STIME=(10:00)

END=(2012/223)

ETIME=(19:00)

The following example shows the TMSSMF control statements that select SMF records from the period 2:00 to 4:00PM on April 15, 2013. You can use the MMMDDYYYY date format on the control statement because DATEFMT=MMMDDYYYY is specified as a JCL parameter. The ACCT= parameter in the following example saves the text RECOVERED BY TMSSMF in the User Accounting field of the volume records to be recovered.

START=(APR152013)

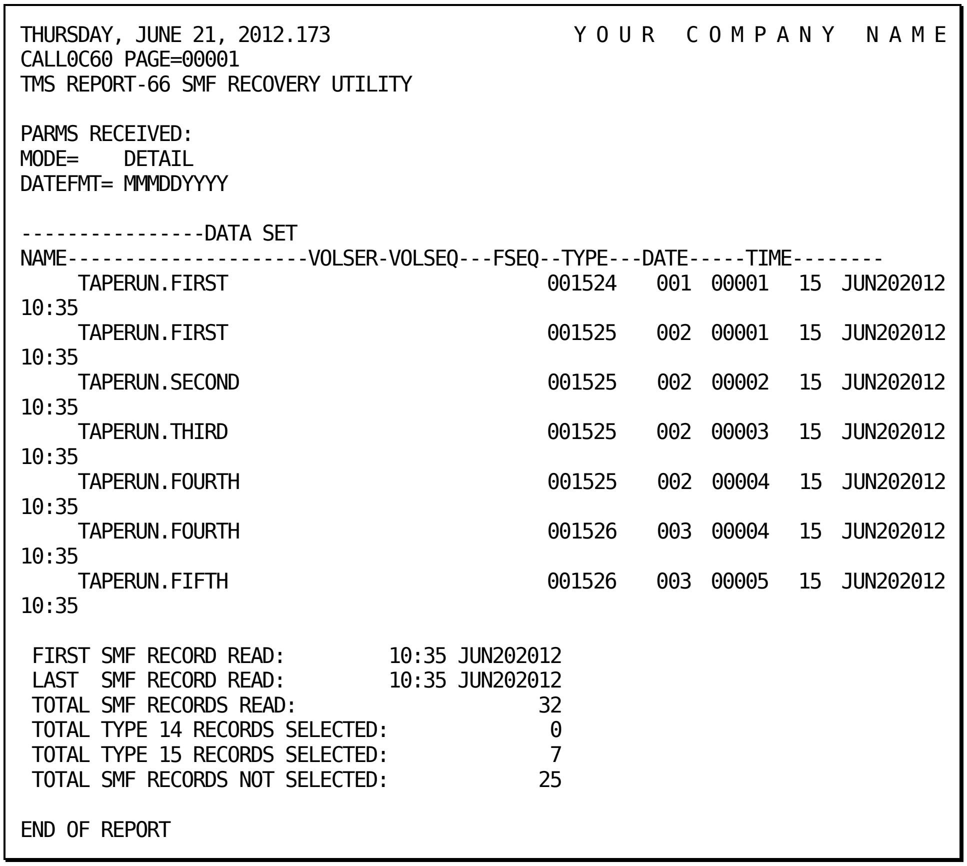
STIME=(14:00)

END=(APR152013)

ETIME=(16:00)

ACCT=’RECOVERED BY TMSSMF’

TMS Report 66 – SMF Recovery Utility Output



```
THURSDAY, JUNE 21, 2012.173                        Y O U R   C O M P A N Y   N A M E
CALL0C60 PAGE=00001
TMS REPORT-66 SMF RECOVERY UTILITY

PARMS RECEIVED:
MODE=    DETAIL
DATEFMT= MMDDYYYY

----------------DATA SET
NAME--------------------VOLSER-VOLSEQ---FSEQ--TYPE---DATE-----TIME--------
     TAPERUN.FIRST                           001524   001  00001   15  JUN202012
10:35
     TAPERUN.FIRST                           001525   002  00001   15  JUN202012
10:35
     TAPERUN.SECOND                          001525   002  00002   15  JUN202012
10:35
     TAPERUN.THIRD                           001525   002  00003   15  JUN202012
10:35
     TAPERUN.FOURTH                          001525   002  00004   15  JUN202012
10:35
     TAPERUN.FOURTH                          001526   003  00004   15  JUN202012
10:35
     TAPERUN.FIFTH                           001526   003  00005   15  JUN202012
10:35

 FIRST SMF RECORD READ:         10:35 JUN202012
 LAST  SMF RECORD READ:         10:35 JUN202012
 TOTAL SMF RECORDS READ:                     32
 TOTAL TYPE 14 RECORDS SELECTED:              0
 TOTAL TYPE 15 RECORDS SELECTED:              7
 TOTAL SMF RECORDS NOT SELECTED:             25

END OF REPORT
```

Note: VOLSEQ is blanks for any Record Type 14 entries.

TMS Report 66 Field Definitions

The TMS report 66 includes the following fields:

TOTAL SMF RECORDS READ

Total number of SMF records that the utility read and examined for tape activity. This number also includes disk data sets processed during the start and end date range and time range.

TOTAL TYPE 14 RECORDS SELECTED

Identifies the number of SMF Type 14 records that are discovered in the given SMFINPUT DD.

TOTAL TYPE 15 RECORDS SELECTED

Identifies the number of SMF Type 15 records that are discovered in the given SMFINPUT DD.

TOTAL SMF RECORDS NOT SELECTED

Total number of SMF records that the utility ignored due to other record type, DASD device, or date and time filters.

TMSSMSUP Utility—Update TMC With SMS Management Classes

TMSSMSUP is a utility which can be used to update the SMS Management Class field in the TMC. For Volume and DSNB records, the field name is SMSMC and is 8 characters.

TMSSMSUP can be executed initially to update the SMSMC field, or when SMS Management Class definitions change. Normally, SMS Management Class names are obtained during OPEN for OUTPUT. SMS Management Class names are used to assign tape scratch pools, enforcing SMS Management Class maximum retention, assigning expiration dates or retention criteria and for other reporting purposes.

SMS Management Classes are obtained by invoking the installed SMS ACS routines by way of the SMS subsystem. Values returned by SMS are used to update the TMC Volume and DSNB records. For more information on coding ACS routines to return Management Class assignments for tape data sets, see the *Programming Guide*.

CA 1 must be active or batch active to run TMSSMSUP.

Report Description

TMSSMSUP outputs TMC Volume and DSNB records to the EARLOUT data set name. This file can be used in a subsequent step with a report writer such as CA Earl and TMSEARL to produce the SMS Management Class Update Report.

Job Control Statements

```
//stepname   EXEC PGM=TMSSMSUP[,PARM='[TEST][,ALL]']
//STEPLIB    DD DSN=CAI.CTAPLINK,DISP=SHR
//SYSPRINT   DD SYSOUT=A
//EARLOUT    DD DSN=smsup.earlout,
//              DISP=(NEW,PASS,DELETE),
//              UNIT=SYSDA,
//              VOL=SER=VOLSER,
//              SPACE=(CYL,5),
//              DCB=(RECFM=FB,LRECL=340,BLKSIZE=n*340)       *NOTE
```

Note: The BLKSIZE may be any multiple of 340.

Parameter Definitions

TEST

Indicates that TMC updates are not performed. The records selected are written to the EARLOUT data set.

ALL

Directs TMSSMSUP to obtain new Management Class names for all active (nonscratch, nondelete) TMC Volume records and associated DSNB records.

If PARM=ALL is not specified, TMSSMSUP will obtain SMS Management Class names for all active (nonscratch, nondelete) TMC Volume records and associated DSNB records that do not currently have a value in the SMSMC field.

Control Statement Specification

There are no control statements for TMSSMSUP.

SMS Update Listing

```
SUNDAY, FEBRUARY 26, 2005.057                      Y O U R   C O M P A N Y   N A M E                        PAGE      1

                                                 PARM=TEST

TMS REPORT-87      SMS UPDATE LISTING

------------------------------------------------------------------------------------------------------------------------------
VOLSER    VOL   FILE  DATA SET NAME                             SMS     CREATE    CREATE        CREATE  EXPIRATION  LAST USE
          SEQ    SEQ                                            CLASS     JOBNAME   DATE              TIME  DATE        DATE
------------------------------------------------------------------------------------------------------------------------------
980000     0      2   CAI.CA1.FILE2.LISTING                     INTERIM   JOBTAPER  YYYY/055    11:36:00  YYYY/060    ZEROS
980001     2      1   CAI.CA1.MVMF.LISTING                      INTERIM   JOBTAPER  YYYY/055    11:36:00  YYYY/056    YYYY/055
980002     3      1   CAI.CA1.MVMF.LISTING                      INTERIM   JOBTAPER  YYYY/055    11:36:00  YYYY/056    YYYY/055

END OF REPORT
```

TMSSMSUP Report Field Definitions

PARM=xxxxxxxx

Parameters passed to program

VOLSER

Volume serial number of tape volume or file

VOL SEQ

Volume sequence of tape volume

FILE SEQ

File sequence of file

DATA SET NAME

Data set name of tape volume or file

SMS CLASS

Management Class of tape volume or file

CREATE JOBNAME

Creating job name of tape volume or file

CREATE DATE

Creation date of tape volume or file

CREATE TIME

Creation time of tape volume or file

EXPIRATION DATE

Expiration date of tape volume or file

LAST USE DATE

Last used date of tape volume

TMSSPLIT Utility—Select Data from the TMC for Use by TMSMERGE

Use the TMSSPLIT utility to perform the first half of a CA 1 split/merge operation, creating an *export* tape of TMC records to be used by TMSMERGE *import* as an *import* data set for another TMC.

TMSSPLIT must be presented with a list of the volume serial numbers that are to be merged into the other TMC. Additional volumes may be included from outside the input list to satisfy TMSSPLIT SCRATCH control statement requirements.

When TMSSPLIT captures a Volume record for a tape containing secondary files, the TMC DSNB records for those data sets are also captured and placed on the export tape. TMSSPLIT also provides the functions of marking selected volumes out-of-area as the export tape is being created, or expiring the Volume records, making them available for scratch processing by TMSCLEAN. The flowchart on Report Description (see page 398) illustrates the TMC Split/Merge operation.

TMSSPLIT requires a list of volume serial numbers as input. Multivolume data sets are automatically accommodated by TMSSPLIT.

CA 1 must be active or batch active to run TMSSPLIT.

Considerations

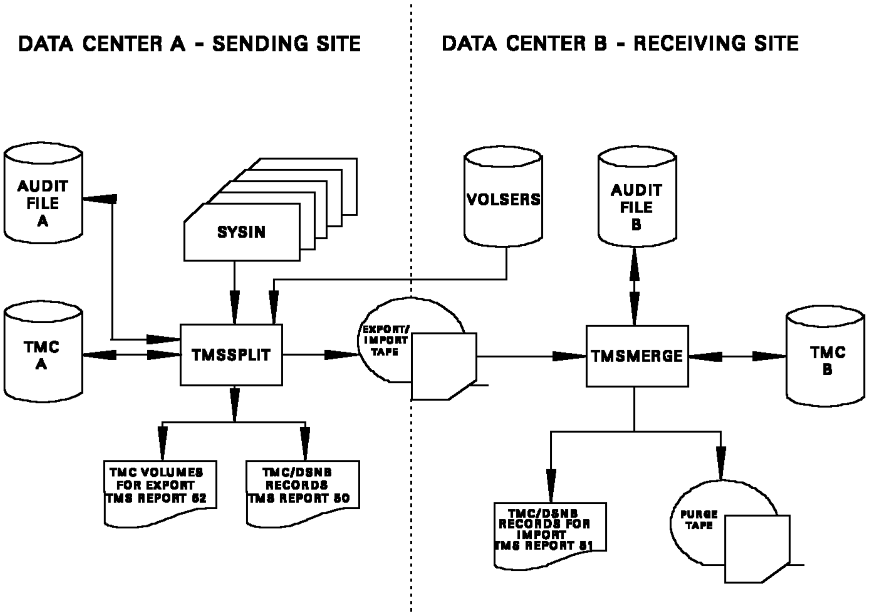
Review the following considerations before beginning the Split procedure.

1. CA 1 Catalog Control and Catalog Days Control of tape data sets (CATALOG, CATLG, CATLG/ddd in control statements and online interfaces, or LABEL=EXPDT=90ddd and LABEL=EXPDT=99000 in JCL) requires that CA 1 keep tapes out of scratch status as long as the data set has an entry in the system catalog. If possible, the system catalog information for these volumes should be made available to the receiving site along with the export tape. If not, you may wish to consider providing a different EXPDT value to the affected records prior to TMSSPLIT execution.
2. If any tapes going to the receiving site are under Vault Management System control, adjustments must be made if the receiving site is using identical vault names (OUTCODEs) as the sending site. Through redefinition of the vault pattern control statements these tapes can be assigned a new vault name or removed from VMS control entirely. Have a copy of the sending site's pattern ready for the receiving site so the patterns can be integrated.
3. If the sending or receiving site uses the Scratch Pool Management feature, the *hlq*.CTAPOPTN members defining the tape pool options need to be built/integrated. Console operators and users should be informed of any visible changes this may cause. The same holds true for the CA 1 options defined in the TMOOPTxx and TMOEDMxx members of hlq.CTAPOPTN, and the Pull data set used by TMSPULL. A Retention Data Set (used by TMSEXPDT) in use at either location also requires similar accommodation.
4. A split/merge operation should not be performed when the sending site's ranges are identical to ranges currently reflecting nonscratch data sets in the receiving site's TMC. The merge operation requires that the Volume records for the import data set be properly formatted in the receiving TMC. They should be in either scratch or inactive (DELETE) status.
5. Before running TMSSPLIT you should run TMSPTRS to insure that there are no problems in the DSNB chains of the ranges to be processed.

Report Description

TMSSPLIT generates two reports: TMS Report 50 - TMC/DSNB Records for Export and TMS Report 52 - TMC Volumes for Export. Samples of these reports appear on TMS Report 50 - TMC/DSNB Records for Export (see page 402) and TMS Report 52 - TMC Volumes for Export (see page 402).

The following diagram illustrates a TMC Split/Merge:


DATA CENTER A - SENDING SITE
DATA CENTER B - RECEIVING SITE
AUDIT FILE A
SYSIN
VOLSERS
AUDIT FILE B
TMC A
TMSSPLIT
EXPORT/IMPORT TAPE
TMSMERGE
TMC B
TMC VOLUMES FOR EXPORT TMS REPORT 52
TMC/DSNB RECORDS TMS REPORT 50
TMC/DSNB RECORDS FOR IMPORT TMS REPORT 51
PURGE TAPE

Split Procedure

Step 1

Execute TMSPTRS and correct any reported multivolume or multi-data set chaining errors. TMSPTRS must run with a condition code of 0 before you continue to the next step.

Step 2

Create a list of volume serial numbers for TMSSPLIT. CA Earl is an excellent utility for creating this list. It provides the capability, for example, to select only those volumes that are not in scratch status and reflect data set names beginning with the characters XYZ, effectively automating the process of deciding which tapes should move.

The list can be created through other means, in any sequential format, as long as the VOLSERS field is in the same offset of each record and is identified to TMSSPLIT with control statements.

Step 3

Stop tape processing and CA 1 batch/online functions on all CPUs sharing the TMC.

Step 4

Execute TMSCOPY to back up the TMC and reset the Audit data set pointers.

Step 5

Execute TMSSPLIT. The output of TMSSPLIT provides information on the number of DSNB and Audit records that is required at the receiving site and a list of every volume selected for the split. This output should be carried to the receiving site along with the export tape. It is recommended that a backup of the export data set be created with IEBGENER or a similar utility.

Step 6

If it becomes necessary to restore the TMC, execute the TMSCOPY utility with PARM='RESTORE,NOAUDIT', using the backup created in Step 4, and return to Step 1. If the ACTION control statement was omitted, no restore operation is needed.

Job Control Statements

```
//stepname EXEC PGM=TMSSPLIT[,PARM='[DATEFMT=(fmt)][,TEST]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//TMSPULL  DD  SYSOUT=A
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(space))                *NOTES
//SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
//SYSOUT   DD  SYSOUT=A
//SORTWK01 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK02 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(TRK,(2,2),,CONTIG)
//EXPORT   DD  DSN=TMS.EXPORT,UNIT=TAPE,DISP=(NEW,KEEP)
//             [,DCB=BLKSIZE=n*366]                          *NOTES
//VOLSERS  DD  DSN=CAI.CA1.LIST,DISP=(OLD,KEEP),UNIT=SYSDA
//SYSIN    DD  *
(control statements here)
/*
```

Note: This space value is the number of volume and DSNB records being split divided by 795.

Note: The BLKSIZE may be any multiple of the LRECL, 366. If no BLKSIZE is specified, the BLKSIZE defaults to 32574 for tape or 8784 for DASD devices.

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Reports 50 and 52. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

TEST

Prevents actual updating of the sending or exporting TMC when the ACTION control statement is used. This parameter is optional.

Control Statement Specification

TMSSPLIT accepts VOLSER, SCRATCH and ACTION control statements.

VOLSER Control Statement

VOLSER is an optional control statement used to specify the position of the volume serial number in the VOLSER data set. The format for the VOLSER control statement is:

```
VOLSER=xxx
```

Control Statement Definition

xxx

Indicates the starting offset position of the volume serial number relative to zero within the VOLSER data set record. Default value is offset 0.

SCRATCH Control Statement

SCRATCH is an optional control statement used to specify the number of scratch tapes required at the receiving site. The format for the SCRATCH control statement is:

```
SCRATCH=xxx
```

Control Statement Definition

xxx

Indicates the number of scratch volumes to be exported. If this number is not satisfied by the volume serial numbers presented as input to TMSSPLIT, then records in scratch status in the sending TMC are selected arbitrarily to satisfy the number specified.

ACTION Control Statement

ACTION is an optional control statement used to specify, in the sending TMC, the disposition of the TMC records selected for export. If the ACTION statement is omitted the TMC record is unchanged. The format for the ACTION control statement is:

```
       { SCRATCH }
ACTION={ outcode }
```

Control Statement Definition

SCRATCH

Indicates that after a TMC record is moved to the extract tape, its expiration date in the TMC is set to the current date, making it eligible for scratch processing by TMSCLEAN.

outcode

Is a one- to four-character out-of-area code. After a TMC record is moved to the export tape, OUTCODE is moved to the OUTCODE field in the TMC. The current date is moved to the OUTDATE field.

TMS Report 50 - TMC/DSNB Records for Export

The first page shows the required number of Audit records and TMC DSNB records in the receiving data sets. The second page lists the data sets that are to be merged.

```
MONDAY, JUNE 20, 2005.177                    Y O U R   C O M P A N Y   N A M E                           ROxxxxx PAGE=00001

TMS REPORT-50 TMC/DSNB RECORDS


**CONTROL CARDS**
VOLSER=000

**IMPORT REQUIREMENTS**
0022733 AUDIT RECORDS (MINIMUM; MORE IF ANY DSNBS OF CORRESPONDING TMC RECORDS ON IMPORT TMC ARE PURGED)
0002858 DSNB RECORDS (MAXIMUM; FEWER IF ANY DSNBS OF CORRESPONDING TMC RECORDS ON IMPORT TMC ARE PURGED)
```

```
MONDAY, JUNE 20, 2005.177                    Y O U R   C O M P A N Y   N A M E                                PAGE=00002

TMS REPORT-50 TMC/DSNB RECORDS

  VOLSER  VSEQ  FSEQ  DATA SET NAME                                JOBNAME   CDATE

  100001  001   001   ADMIN.PAYR.DLY.G0406V00                      ADMINPDY  MMMDD YYYY
  100002  001   001   ADMIN.PAYR.WLY.G0088V00                      ADMINPWY  MMMDD YYYY
  100003  001   001   ADMIN.PAYR.DLY.G0405V00                      ADMINPDY  MMMDD YYYY
  100004  001   001   ADMIN.PAYR.WLY.G0087V00                      ADMINPWY  MMMDD YYYY
  100005  001   001   ADMIN.PAYR.DLY.G0404V00                      ADMINPDY  MMMDD YYYY
  100007  001   001   ADMIN.PAYR.DLYBKUP.G0403V00                  ADMINPDY  MMMDD YYYY
  100009  001   001   ADMIN.PAYR.DLY.G0402V00                      ADMINPDY  MMMDD YYYY
  100010  001   001   ADMIN.PAYR.WLY.G0086V00                      ADMINPWY  MMMDD YYYY


         END OF REPORT 50  -  TOTAL ITEMS LISTED=        8
```

TMS Report 52 - TMC Volumes for Export

This example lists the volume serial numbers to be exported and merged into the other TMC.

```
MONDAY, JUNE 20, 2005.177                    Y O U R   C O M P A N Y   N A M E                           ROxxxxx PAGE=00001

TMS REPORT-52 TMC VOLUMES FOR EXPORT     * INDICATES CHAINED VOLUMES

  VOLSER   100001,  100002,  100003,  100004,  100005,  100007,  100009,  100010,  100011,  100012,
           100013,  100015,  100016,  100017,  100019,  100021,  100023,  100025,  100027,  100031,
           100033,  100035,  100037,  100045,  100046,  100047,  100048,  100050,  100063,  100064
```

TMSSPLIT Report Field Definitions

****CONTROL CARDS****

Input control statements

AUDIT RECORDS

Estimated number of Audit records required

DSNB RECORDS

Estimated number of DSNB records required

VOLSER

Volume serial number of tape volume or file or list of tape volumes written to export

VSEQ

Volume sequence number of tape volume

FSEQ

File sequence number of file

DATA SET NAME

Data set name of tape volume or file

JOBNAME

Creating job name of tape volume or file

CDATE

Creation date of tape volume or file

TMSSTATS Utility—Report Active System Options

Use the TMSSTATS utility to report on the internal control tables as defined by *hlq*.CTAPOPTN and currently resident in the operating system.

CA 1 must be active or batch active to run TMSSTATS.

Report Description

TMSSTATS generates the following reports, and each report has a heading which reflects the name and contents of the *hlq*.CTAPOPTN member from which the internal form was built. Samples of these reports are provided.

- Options Table Status Report (TMS Report 54) (see page 405)
- CA Message Intercept Table Status (TMS Report 55) (see page 406)
- Scratch Pool Definition Report (TMS Report 56) (see page 407)
- Scratch Pool Assignment Report (TMS Report 57) (see page 407)
- EDM Rule Assignment Report (TMS Report 61) (see page 407)
- Keyword Table Report (TMS Report 67) (see page 408)
- Security Table Report (TMS Report 68) (see page 408)

Job Control Statements

```
//stepname EXEC PGM=TMSSTATS
//            [,PARM='[DATEFMT=(fmt)][,OPT][,MSG][,SCR][,NSM][,EDM][,ALL]'][,KEY]
[,SEC]
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
```

Parameter Definition

DATEFMT=(fmt)

(Optional) Specifies the date format on reports that TMSSTATS generates. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses.

OPT

Indicates that the Options Table Status Report is printed.

MSG

Indicates the CA 1 Message Intercept Table Status is printed.

SCR

Indicates that the Scratch Pool Definition Report is printed.

NSM

Indicates that the Scratch Pool Assignment Report is printed.

EDM

Indicates that the External Data Manager (EDM) Rule Assignment Report is printed.

KEY

Indicates that the Keyword Table Report is printed.

SEC

Indicates that the Security Table Report is printed.

ALL

Indicates that all CA 1 internal table reports are printed. This option is the default.

More information:

Override the Preferred Date Pattern (see page 21)

TMS Report 54 - Options Table Status Report

```
WEDNESDAY, JUNE 22, 2005.157                     Y O U R   C O M P A N Y   N A M E                             ROxxxxx PAGE=00001
 CA-1/MVS GENLEVEL 9602L052
TMS REPORT-54                          OPTIONS TABLE STATUS REPORT
                    OPTION KEYWORD                      VALUE

                       TMCDSN                           ASM.CA1DV.TMC2X
                       AUDDSN                           ASM.CA1DV.AUDITX
                       DSN                              TMSDISP
                       RC                               000
                       R9                               000
                       ABE                              003
                       RR                               000
                       TW                               0009
                       DCHG                             YES
                       LCHG                             YES
                       NLWTOR                           YES
                       TCHG                             YES
                       LAB                              YES
                       RO                               YES
                       UNCATA                           NO
                       CYD                              NO
                       MIXEXP                           WARNING
                       CDAY                             TAPE
                       CJOB                             DSN
                       RP                               0000000005
                       MG3480                           YES
                       REDM                             000
                       THDSNB                           0090
                       THAUDT                           0080
                       KEYTAP                           LDATE/005
                       DATEFMT                          MMMDD YYYY
                       OCEOV                            NO
                       DSNB                             NO
                       CATSEC                           NO
                       YSVC                             YES
                       BATCH                            NO
                       PSWD                             YES
                       CMD                              YES
                       FUNC                             NO
                       SCRTCH                           YES
                       UNDEF                            ALLOW
                       CREATE                           UPDATE
                       WRKFLS                           YES
                       DSEALL                           NO
                       FORNDSN                          ALL
                       UX0AUPD                          NO
                       XCLN1                           NO
                       XCLN2                           NO
                       XCTLG                           NO
                       XCYCL                           NO
```

```
WEDNESDAY, JUNE 22, 2005.157                     Y O U R   C O M P A N Y   N A M E                              PAGE=00002

                       XEXP                               NO
                       XOCAT                              NO
                       XTPNT                              NO
                       XTPPR                              NO
                       XVLT1                              NO
                       SMS                                 YES
                       SR                                  NO
                       RV                                  030
                       QSAMBF                              25
                       MSGPFX                              CTS
                       SCRCAT                              GDG
                       SECWTO                              NO
                       LOGSVC                              ALL
                       AUDB4                               BATCH
                       RECRE8                              KEEP
                       ADGDG                               NO
                       XSCR                               NO
                       MIXGDG                              NO
                       OCTLG                               NO
    ***  B1 SECURITY IS NOT ACTIVE  ***
 END OF OPTIONS TABLE STATUS REPORT
```

TMS Report 55 - CA 1 Message Intercept Table Status

```
WEDNESDAY, JUNE 22, 2005.157                     Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

                            CTS MESSAGE INTERCEPT TABLE ENTRIES CURRENTLY ACTIVE
             MESSAGE           PRODUCT IDENTIFICATION
             IAT5210           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IAT5624           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IEC101A           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IEC501A           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IEC501E           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IEC502E           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IEC507D           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IEC534D           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IEC704A           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IEF233A           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IEF233D           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             IEF234E           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             JBB101A           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             JBB501A           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             JBB501E           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             JBB502E           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             JBB507D           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             JBB534D           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             JBB704A           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             JDJ233A           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             JDJ233D           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
             JDJ234E           CA 1    (TMS) TAPE MANAGEMENT SYSTEM
```

TMS Report 56 - Scratch Pool Definition Report

```
WEDNESDAY, JUNE 22, 2005.157                     Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

                        SCRATCH POOL DEFINITION REPORT FOR CA-1    (TMS) TAPE MANAGEMENT SYSTEM
         SCRATCH POOL(S) DEFINED IN CTAPOPTN MEMBER: TMOSCRCC
          SCRATCH POOL ID          VOLUME SERIAL RANGE(S)

           TEMP-TAPEPOOL            000001-000010
           TEST??-                  000020-000021, 000022-000022
           WHOPOOL-                 000011-000011, 000012-000013, 000014-000015
           PATTPOOL                 980001-980005
           VOLPOO                   TDI072-TDI079
           MORPOO                   TDI080-TDI084
           EDMVOL                   TDI070-TDI071
           OTHER-TAPES              TDI085-TDI089
           WITT                     TDI045-TDI049
 END OF SCRATCH POOL DEFINITION REPORT
```

TMS Report 57 - Scratch Pool Assignment Report

```
WEDNESDAY, JUNE 22, 2005.157                     Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

                        SCRATCH POOL ASSIGNMENT REPORT FOR CA-1    (TMS) TAPE MANAGEMENT SYSTEM
         NON-SPECIFIC MOUNT RULES ASSIGNED IN CTAPOPTN MEMBER: TMONSMBL
MANAGEMENT CLASS / EDM NAME / DATASET NAME                         JOBNAME      LOW DATE   HIGH DATE  UNIT SCRATCH POOL

MGMTCLAS=BACKUP                                                    N/A          N/A        N/A        N/A  OTHER-TAPES
EDM=HSM                                                            N/A          N/A        N/A        N/A  EDMVOL
EDM=EDMM                                                           N/A          N/A        N/A        N/A  EDMVOL
A????                                                              -            N/A        N/A        -    PATTPOOL
SSDDEV.CA?.LOAD                                                    -            0031       0040       -    PATTPOOL
*.XYZ*                                                             -            N/A        N/A        -    PATTPOOL
ABC-XYZ                                                            -            N/A        N/A        -    PATTPOOL
FRAN-                                                              -            USER/003   USER/003   -    TEMP-TAPEPOOL
FRAN-                                                              -            MMMDD YYYY MMMDD YYYY -    OTHER-TAPES
FRAN-                                                              -            0011       0030       -    TEMP-TAPEPOOL
FRAN-                                                              -            0010       0010       -    TEMP-TAPEPOOL
FRAN-                                                              F-           N/A        N/A        -    OTHER-TAPES
PRODR-                                                             JOB-         N/A        N/A        -    MORPOO
TESTR-                                                             JOB-         N/A        N/A        -    VOLPOO
 END OF SCRATCH POOL ASSIGNMENT REPORT
```

TMS Report 61 - EDM Rule Assignment Report

```
WEDNESDAY, JUNE 22, 2005.157                     Y O U R   C O M P A N Y   N A M E                               ROxxxxx PAGE=00001

                          EDM RULE ASSIGNMENT REPORT FOR CA-Tape Beta  TAPE MANAGEMENT SYSTEM
         EXTERNAL DATA MANAGER RULES ASSIGNED IN CTAPOPTN MEMBER: TMOEDMXX
  EDMID    DATA SET NAME                                 PROGRAM NAME   JOBNAME NAME   DD NAME

  EDMM     ASM2.-                                        -              -              -
  HSM      HSM.-                                         -              -              -
  XX       -                                             -              -              OUT1A
  YY       -                                             -              -              OUT2B
  EDMB     -                                             -              -              OU01SL02
  EDMA     -                                             -              -              OU02SL01
  DMS      -                                             ADSMI002       -              -
  HSM      -                                             ARCCTL         -              -
  WSF      -                                             AREPVT00       -              -
 END OF EDM RULE ASSIGNMENT REPORT
```

TMS Report 67 – Keyword Table Report

```
THURSDAY, MAY 08, 2014.128                      Y O U R   C O M P A N Y   N A M E                                 ROxxxxx PAGE=00001

TMS REPORT-67                          CA 1 KEYWORD TABLE ENTRIES CURRENTLY ACTIVE

KEYWORD  SHORT KWD  OFFSET  LENGTH  XCOORD  YCOORD  FLAG

USER123  U          10      02      00      00      MUL

ROBOT_TYP_50=ROBOT

ROBOT_TYP_150=VROBOT
```

TMS Report 68 – Security Table Report

```
THURSDAY, MAY 08, 2014.128                     Y O U R   C O M P A N Y   N A M E                                 ROxxxxx PAGE=00001
 TMS REPORT-68                CA 1 SECURITY TABLE ENTRIES CURRENTLY ACTIVE FROM MEMBER TMOSEC00

 PROFILE=SYSPROG
 TINQ=(ALL)
 TUPD=(ALL)
 TUNCT=(ALL)
 TECUR=(ALL)
 DUNCT=(ALL)
 DECUR=(ALL)
 PROFILE=USERINQ
 TINQ=(TMC,DSNB)
 TUPD=(BLK)
 TUNCT=(LBL)
 TECUR=(BLK)
 DUNCT=(BLK)
 DECUR=(BLK)
 PROFILE=USERUPD
 TINQ=(TMC,DSNB)
 TUPD=(TMC,DSNB)
 TUNCT=(EXP,EXT,RET,LBL)
 TECUR=(EXPDT)
 DUNCT=(ALL)
 DECUR=(EXPDT)
 PROFILE=LIBRARY
 TINQ=(ALL)
 TUPD=(TMC,DSNB)
 TUNCT=(EXP,EXT,ADD,DEL,CHE,CLE,RET,CHO,LBL)
 TECUR=(DSN17,SLOT,SLOTS,EXPDT,FLAG1,CLNCNT,OUTDATE,OUTCODE,USECLN,DATECLN,TRERRC,TWERRC)
 TECUR=(PRERRC,PWERRC)
 DUNCT=(ALL)
 DECUR=(EXPDT)
 END OF SECURITY TABLE REPORT
```

TMSSTATS Report Field Definitions

The TMS reports contain the following fields:

OPTION KEYWORD

System option keyword

VALUE

Assignment for system option keyword

MESSAGE NUMBER

Operating system message prefix

SEQ

Operating system message prefix sequence number

PRODUCT IDENTIFICATION

Product processing for MVS message prefix

SVC NO.

(DEC)

CA 1 SVC decimal value

(HEX)

CA 1 SVC hexadecimal value

SERVICE CODE

Processing service code

SCRATCH POOL ID

Scratch pool name

SEQ

Scratch pool sequence number

VOLUME SERIAL RANGE(S)

Volume ranges assigned to scratch pool

DATASET NAME

Data set name qualifier

JOB NAME

Job name qualifier

SCRATCH POOL

Scratch pool name

EDMID

External Data Manager ID

DATA SET NAME

Data set name qualifier

PROGRAM NAME

Program name qualifier

JOBNAME NAME

Job name qualifier

DD NAME

DDname qualifier

TMSTPNIT Utility—Initialize Tape Volumes

Use the TMSTPNIT utility to initialize tape volumes as SL, NL or AL.

TMSTPNIT verifies your request to initialize a tape volume by checking the TMC and, when possible, checking the tape label.

Important! Nonlabeled tapes and those which have not been written upon are considered valid for initialization. Standard OPEN and CLOSE are issued to process the tapes. Thus, SMS, CA Top Secret® Security, RACF, CA 1 and any other components or products receiving control during OPEN/CLOSE processing may be invoked. In an MVS environment, APF authorization and BLP authorization are required for TMSTPNIT and the modules it calls: CTSPARSE, CTSSIO, CTSTIO, TMSPRINT, and TMSSVC.

For any tape cartridges where a VOL1 record is present (this includes AL and SL labeled tapes), the VOLSER is obtained from the tape. For round tapes, the VOLSER can be obtained only when the tape is known to contain data (OLDTAPE), since any attempt to read a blank tape will cause the drive to search the entire reel for data. In all cases (except where the hardware indicates the tape contains no data), when a VOLSER cannot be obtained from the tape, the operator is prompted for one. To determine if the mounted tape is valid for initialization, the status of the obtained VOLSER in the TMC, the input command (INTAPE, NEWTAPE, OLDTAPE or OUTTAPE) and user exit (if any) is checked.

TMSTPNIT opens tapes with BLP input. Some security packages software protect the device if the user does not have BLP write authority to issue a CCW = x'7782??80'. This CCW causes the device to treat the tape as write-protected resulting in a write-protect error message from TMSTPNIT. In the case of when the user does not have BLP read authority, the system (without any indication or warning) treats BLP as NL. This can manifest itself either by successive mount/dismounts when the tape is blank or unreadable or in IECTMS1 messages. TMSTPNIT *must* be executed under a user ID which has read/write BLP authority.

The method of volume verification for TMSTPNIT differs depending on the request and type of device. The overriding rule is that the operator is prompted any time that TMSTPNIT cannot obtain sufficient information from the device to ensure that a *live* tape is not being overwritten. The procedure differs depending on whether the tape is being opened for input or output. Cartridges are always opened for input; reels are opened for output unless OLDTAPE is specified (opening up a blank reel for input will scan off the end of tape for data).

When a tape is opened for output with BLP, TMS intercepts the open and issues an IECTMS1 WTOR prompting the operator for the VOLSER of the tape that is mounted to ensure that a *live* tape is not over written. This is because once a tape has been opened for output, closing it will result in end-of-file marks being written which will destroy data. When OLDTAPE is specified for a reel, the user is indicating that the tape contains data; thus, the tape can be opened for input and the software can attempt to read an 80 character VOL1 record from the tape. If this read is successful, the software can determine the true VOLSER of the tape and the TMC can be checked to ensure that this is not a *live* tape. If there is no VOL1 record (NL) on the tape, a CAL0TN01 WTOR is issued. When a cartridge is being processed, the device can recover from attempting to read a blank tape (new, not erased) and provide sense data from the device to indicate that this is a blank tape not a *live* tape, so no verification message is issued. In most cases, when attempting to read a valid tape which can not be processed by this device (such as a 3480 with a 36 track tape), the device will return the VOLSER in the format 24 sense data and the TMC can be checked using this VOLSER without operator intervention. In all other cases when an error occurs attempting to read the VOL1 record (such as may be the case with a bulk erased tape) or no VOL1 record is present, a CAL0TN01 WTOR is issued.

TMSTPNIT uses both BLP and 98000 processing. Therefore, the proper security access must be set up to run TMSTPNIT. For more information see the *Programming Guide*.

CA 1 must be active or batch active to run TMSTPNIT.

User Exit

An optional user exit TMSXTPNT is provided for tailoring TMSTPNIT processing. It is invoked prior to each tape being processed and is called only if the option UXTPNT is set to the name of the exit itself in the TMOOPTxx member of *hlq*.CTAPOPTN. TMSXTPNT can continue processing, reject the tape, or modify the TMC record or tape labels.

Note: For more information on this user exit, see the *Programming Guide*.

Report Description

TMSTPNIT generates TMS Report 41 - Tape Initialization. A sample of this report appears on TMS Report 41 - Tape Initialization (see page 417).

Job Control Statements

```
//stepname EXEC PGM=TMSTPNIT
//             [,PARM='[DATEFMT=(fmt)][,TEST][,VERIFY=(opt)]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//* NOTE - THE ABOVE LIBRARY MUST BE IN IEAAPFxx or PROGxx
//SYSPRINT DD  SYSOUT=A
//LABELDD  DD  UNIT=(tape,,DEFER)
//SYSIN DD  *
(control statements here)
/*
```

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 41. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

TEST

The TEST parameter allows you to simulate the execution of TMSTPNIT without actually modifying the contents of the tapes or the TMC. This can be done with or without operator intervention. Operator intervention is determined by the presence of a DD card for LABELDD. (In round tapes, the operator intervention can occur only when the OLDTAPE command is requested. This is because of problems opening a blank round tape for input.)

When there is no LABELDD DD statement present, no operator intervention is required and console messages are kept to a minimum. Control statements are read, checked for syntax and valid operands. The SER operand is checked against the contents of the TMC (except when OUTTAPE is specified), and the user exit is called.

When the LABELDD DD statement is present in the JCL, the requested tapes are actually mounted (except for round tapes which are not mounted unless OLDTAPE is specified) and additional edits are performed using the label information contained on the tape. All console messages including those requiring responses are issued.

VERIFY=(opt)

The VERIFY parameter determines when, and if, label verification messages are issued for cartridges. It is valid for all cartridges, ignored for reels unless the OLDTAPE command is used, and not applicable when the test option is specified without a DD card for LABELDD. The verify command can have one of the following values:

ERROR

(Default) - the CAL0TN01 message is issued only for tapes not containing a VOL1 record (NL).

CHANGE

The CAL0TN01 message is issued whenever a tape containing data is having its VOLSER changed or for tapes not containing a VOL1 record (NL), *NL*.

ALWAYS

A CAL0TN01 message is always issued. Reels will always display '******' for the VOL1 record's volser if OLDTAPE is not specified. For example, no attempt is made to read the VOL1 record. New tapes will display '*NEW*' for the VOL1 record's VOLSER (cartridges only, OLDTAPE is not valid for new reels and will run the tape off the end of the reel if specified for a never used tape).

ABEND

Task will abend after issuing a CAL0TN02 or CAL0TN03 message without waiting for an operator response.

ROBOT

Operator is not prompted for volume verification (CAL0TN01 message) for NL tapes.

JCL Considerations

LABELDD

This defines the tape drive that is used to initialize volumes.

tape

Specifies a tape device used to initialize tape volumes. The DEFER keyword is required.

Completion Codes

Possible completion codes and their meanings are listed below.

0

Successful completion.

4

Successful completion with operator intervention. For example, a live tape was mounted for initialization, it was rejected and a valid tape was mounted and successfully initialized. Another example would be a write-protected tape was mounted, rejected, and non-write-protected tape was mounted and successfully initialized. You should check the console log for specifics.

8

All tapes were not successfully initialized. For example, a specific SER was not in the TMC or had the wrong status. You should check the TMS Report 41 (and possibly the console log) for specifics.

12

Invalid input, such as a syntax error or invalid JCL parameter. You should check the TMS Report 41.

16

Missing DD statement.

Note: The return code is the highest code encountered, and does not preclude the existence of errors having lower return codes, (RC=12 does not mean that there were no errors that would not have produced an RC=8). For example, if you receive an RC=8 you should also check the console for RC=4.

Control Statement Specification

The control statement indicates if the volume is under CA 1 control, specifies the volume serial number(s) to be used in the initialization, and describes the current status of the volume in the TMC. The control statements are free-form and may span to as many statements as required but cannot exceed position 71. Comments in the form /*xxxxxxx*/ may appear in the control statements.

```
1

verb SER=vvvvvv[,OWNER='cccccccccc[cccc]'][,NUMBTAPE=n][,LABTYPE=AL]
               [,DISP={UNLOAD}][,ACCESS=x] [,ERASE]
               [      {REWIND}]
```

Control Statement Definition

verb

Specifies if the volume serial number is under CA 1 control. It must begin in position 1 and must be followed by a space. The values for verb are:

NEWTAPE

Specifies that the volume serial number is under CA 1 control and the physical tape volume is to be replaced. The TMC record for the volume must be in *scratch* status. The tape's birth date field is updated to reflect the date on which the tape was initialized, and all eight read/write error count fields are set to zero. If the tape is marked bad in FLAG3, this is turned off once the tape is reinitialized. Tapes in inactive (delete) status are not valid for this option.

OLDTAPE

Specifies that the volume serial number is under CA 1 control and the tape volume needs to be reinitialized. The TMC record for the volume must be in *scratch* status.

When a tape has been successfully initialized using the OLDTAPE verb, the TMC data set name for the volume is updated to reflect that the volume has been initialized, and the volume is available for use as a scratch tape. The four fields containing read/write error counts since the last cleaning is set to zero (read/write error counts since initialization is not cleared). If the tape is marked bad in FLAG3, this is turned off once the tape is reinitialized.

OUTTAPE

Specifies that the volume serial number to be initialized is not a CA 1 controlled tape volume (no Volume record in the TMC with a matching VOLSER). When the OUTTAPE verb is used to initialize a tape volume, an Audit record is written to reflect the volume initialized.

Note: Only one tape is initialized for this verb regardless of NUMBTAPE (see description on following page).

INTAPE

Specifies that the volume serial number to be initialized is under CA 1 control and the TMC record for the volume is in inactive, (DELETE) status, and in scratch status.

When a tape volume has been successfully initialized using the INTAPE verb, the TMC data set name is updated to reflect that the volume was initialized. In addition, the birth date field in the TMC record is updated with the current date and the volume is available for use as a scratch tape. TMSTPNIT will set all eight read/write error count fields to zero.

SER=vvvvvv

Specifies the volume serial number of the first or only tape to be labeled. The value of vvvvvv can be incremented by one for each additional tape to be initialized. When processing multiple tapes, TMSTPNIT starts with the specified SER= keyword and continues with the next volume that appears in the TMC for the NUMBTAPE volume. Volumes which are not in the correct status are reported and skipped. If the volume serial number is less than 6 characters, it must be enclosed in single quotes.

OWNER='cccccccccc[cccc]'

Specifies the owner's name or similar identification. The information is specified in character format. It can be up to 10 characters in length for EBCDIC and BCD volume labels, or up to 14 characters in length for volume labels written in ASCII. If this name contains blanks or other special characters (except periods or hyphens), it must be enclosed in single quotes. Embedded quotes may be indicated by specifying two single quotes. If not specified, the VENDOR field from the TMC is used. If specified, the first 8 characters are saved in the TMC VENDOR field.

NUMBTAPE=n

The value of n can be from 1 to 999. If omitted, one tape is initialized. Otherwise initialization is attempted of each tape in the TMC, starting with the one specified by SER= and continuing for n tapes. n will include tapes that are invalid or fail.

Note: If OUTTAPE is specified, only one tape is initialized regardless of NUMBTAPE.

LABTYPE=xx

Specifies the type of LABEL= to be written to the tape. SL is IBM standard label, AL is ANSI standard label, and NL is no label (tapemark written).

DISP=

Specifies whether to UNLOAD or REWIND the tape after initialization. UNLOAD is the default.

ACCESS=c

Single character for ANSI security access. Valid values are blank or A thru Z. This is valid only if LABTYPE= has been specified as AL.

ERASE

Issues a data secure erase to destroy any residual data on the tape (may significantly increase elapsed time).

Control Statement Examples

The following control statement will initialize a new tape with volume serial number 012345:

```
NEWTAPE SER=012345
```

To initialize five new tapes with the first five available (deleted) TMC volume serial numbers, use the following:

```
INTAPE NUMBTAPE=5,SER=012345
```

Note: In an MVS environment, APF authorization and BLP authorization are required for TMSTPNIT and the modules it calls: CTSPARSE, CTSSIO, and CTSTTIO.

TMS Report 41 - Tape Initialization

```
TMS REPORT-41                                TMSTPNIT INITIALIZATION REPORT                                 YYYY/044

 PARMS RECEIVED: VERIFY=CHANGE

 NEWTAPE SER=TST001,DISP=REWIND,OWNER='TAPE TESTERS'
 VOLUME=TST001  LABEL-TYPE=SL  ACCESS=        OWNER=TAPE TESTERS                TAPE INITIALIZED

 OLDTAPE SER=TST002,DISP=REWIND,OWNER='TAPE TESTERS',NUMBTAPE=4
 VOLUME=TST002  LABEL-TYPE=SL  ACCESS=        OWNER=TAPE TESTERS                TAPE INITIALIZED                    WAS 980046
 VOLUME=TST003  LABEL-TYPE=SL  ACCESS=        OWNER=TAPE TESTERS                TAPE INITIALIZED
 VOLUME=TST004  LABEL-TYPE=SL  ACCESS=        OWNER=TAPE TESTERS                TAPE INITIALIZED
 VOLUME=TST005  LABEL-TYPE=SL  ACCESS=        OWNER=TAPE TESTERS                TAPE INITIALIZED

 OUTTAPE SER=X00001,DISP=REWIND
 VOLUME=X00001  LABEL-TYPE=SL  ACCESS=                                          TAPE INITIALIZED                    WAS 980046

 INTAPE SER=TST006,DISP=REWIND,OWNER='TAPE TESTERS'
 *** TST006 ---- INITIALIZATION BYPASSED ***                                    TAPE MUST BE IN DELETE STATUS
```

TMSTPNIT Report Field Definitions

Control statement image

Input control statement

VOLUME=xxxxxx

Volume serial number of tape to be reinitialized

LABEL-TYPE=xx

Type of label to be written on tape

ACCESS=x

ANSI security access code to be written on tape

OWNER=xxxxxxxx

Owner information to be written on tape

Message

Processing message for initialization requests

TMSTPPRO Utility—Process Tape Volume Header Label Information

Use the TMSTPPRO utility to perform the following functions:

- Verifies tape header labels.
- Constructs TMC records on the TMC from tape header label information and can be used to build the initial TMC.
- Erases sensitive data from scratched tape data sets.
- Constructs TMC records on an output file from tape header label information.

When executed, TMSTPPRO will request, and the system will issue, an initial mount request. Thereafter, TMSTPPRO will continue to make mount requests until a system STOP command is issued for the job. Once the STOP is issued, the last mount of a volume will terminate the job.

For each tape mounted, TMSTPPRO reads the tape header label, constructs a TMC formatted record from the tape header label information, then performs the processing requested by the user in the PARM field.

If no expiration date is supplied in the tape header label or by the macro, and none is supplied by the EXPDT parameter, the expiration date is set to 1960/001 and the scratch bit is turned on.

Multivolume chaining is not performed. However, creation date and time are filled in; multivolume tapes should be mounted to ensure correct chaining at a later time.

TMSTPPRO modifies the JFCB label type to BLP.

Note: HDR1 tape labels contain only the last 17 characters of the data set name. Catalog Control EXPDTs do not function for TMC records created by TMSTPPRO with data set names greater than 17 characters.

Console messages may be issued during TMSTPPRO processing. These messages are documented in the *Message Reference Guide*.

Depending on the function to be performed, CA 1 may be active, batch active or inactive to run TMSTPPRO.

Note: This program is not recommended for usage when the tapes to be read are in a robotic type of device, such as IBM 3494/3495 or STK silo.

User Exit

An optional user exit TMSXTPPR is provided for tailoring TMSTPPRO processing. It is invoked prior to each tape being processed and is called only if the option UXTPPR is set to the name of the exit itself in the TMOOPTxx member of *hlq*.CTAPOPTN. TMSXTPPR can continue processing, reject the tape for processing, or abend.

Note: For more information on this user exit, see the *Programming Guide.*

Report Description

TMSTPPRO does not produce a report.

Job Control Statements

```
//stepname EXEC PGM=TMSTPPRO[,PARM='[DATEFMT=(fmt)],parm']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//IEFRDER  DD  UNIT=(TAPE,,DEFER),DISP=(OLD,KEEP),LABEL=(,BLP),
//             VOL=SER=anyvol,DSN=any.tape.file
//OUTPUT   DD  DSN=CAI.CA1.IMAGE,UNIT=TAPE,DISP=(NEW,KEEP)     *NOTE
```

Note: The DD for output is necessary for an output request, for example, PARM=OUTPUT.

Parameter Definition

More than one parameter may be specified.

The parameters specified are processed in the order shown, not in the order of their appearance in the EXEC statement.

DATEFMT=(fmt)

Specifies the date format to be used by TMSTPPRO. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

CONSOLE

Reads the tape header label and formats a display to the console operator's terminal through route codes 3 and 5 (TAPE POOL/TAPE LIBRARY). This option is the default.

COMPARE

Reads the tape header label and compares the data set name of the tape to that of the TMC. Issues a WTO of the tape header and the TMC data set name if a difference exists. If the TMC data set name is HEXZEROS, it issues a message. If the data set names are equal, it displays the contents of the tape header label.

UPDATE [,KEYWORD][,EXPDT=[date or keyword]]

Reads the tape header label and constructs a dummy TMC record. Directly updates the TMC only if the TMC record is in scratch status (the scratch bit is on) and the TMC expiration date (EXPDT) is less than or equal to today's date.

KEYWORD

Indicates that expiration dates in the range of 98001 to 99364 are to be treated as CA 1 keywords.

EXPDT=[date or keyword]

Assigns this date or keyword as the TMC expiration date (EXPDT) of the tape. The tape is now eligible for Retention Data Set processing under TMSEXPDT. The highest expiration date allowed is 2155/366.

ERASE

Reads the tape header label and verifies that the tape is in the TMC, is in scratch status, and that the data set names match. If the volume serial number was not supplied by VOLLABI (the VOLSER from the VOL1 record on SL or AL tapes) or the TMSXTPPR user exit, the console operator is asked to supply the volume serial number. The final action taken before the tape is erased is to verify with the console operator that the erase is still desired on this volume.

This verification may be requested or suppressed as a subparameter of the ERASE option. For example, ERASE(VERIFY) will prompt the console operator for final approval. ERASE(NOVERIFY) will suppress the console operator's intervention. The default is VERIFY.

The position of the tape, before the erase sequence is to begin, will depend on the absence or presence of the volume serial number at the time the erase logic was invoked. If the console supplied the volume serial number, the tape is rewound and ERASE will start at load point. If the volume serial number was supplied by the tape header (SL), or supplied by the macro (NSL), the tape is not rewound.

In the ERASE sequence, a tape mark is written followed by an attempt of IBM's Data Security Erase (DSE). If the DSE fails at any point, the DSE is dropped and an ERASE gap loop is used in its place.

OUTPUT

Reads the tape header label and constructs a TMC record that is written to a sequential output file. If ERASE is specified and fails, the output record is not written.

Control Statement Specification

There are no control statements for TMSTPPRO.

TMSUDSNB Utility—Apply Batch Updates to DSNB/Control Records

Use the TMSUDSNB utility to update or verify any field in TMC Control Records or DSNB records.

CA 1 must be active or batch active to run TMSUDSNB.

If more than one type of expiration date is found in a volume when a DSNB is created, a warning message is issued if MIXEXP is set to WARNING in member TMOOPTxx of *hlq*.CTAPOPTN during installation. The warning message is:

IEBTMS10 - WARNING A MIXED EXPIRATION WAS ISSUED FOR vvvvvv

Important! Only systems programmers responsible for maintaining CA 1 should use this program.

You can also activate or deactivate the UNIQUE_RNAME feature by turning the x'40' bit ON (activate) or OFF (deactivate) in FLAG1 of CTL#1. Before you change this indicator (ON to OFF or OFF to ON), stop tape processing on all systems that share the TMC. After you change the flag setting, run TMSINIT on all active systems that share the TMC. Then allow tape processing to continue.

If the RNAME bit in FLAG1 x'40' is changed from on to off or from off to on, the return code changes from 00 to 04 and the following warning message is issued:

***** RNAME FUNCTIONALITY HAS BEEN MODIFIED - TMSINIT REQUIRED ON ALL SYSTEMS *****

If you use an unextended TMC and attempt to turn on the RNAME bit in FLAG1 (x'40'), the processing stops with the return code 12 and the following message:

***** 'FLAG1' CANNOT BE CHANGED**

Important! When multiple LPARs share the TMC or if tape processing is active on more than one system, activate or deactivate the UNIQUE_RNAME feature using the TMSXTEND utility.

Note: Verify if all systems sharing the TMC are at the appropriate maintenance level to allow the UNIQUE_RNAME feature to be enabled. Run TMSAUDEX and validate that all systems sharing the TMC have a "Y" under the URC column in TMS-REPORT-81.

More information:

TMSXTEND Utility—Dynamic TMC Extend Process Control Program (see page 472)

Report Description

TMSUDSNB produces TMS Report 48 - TMC DSNB and CTL Updates Report. A sample of this report appears on TMS Report 48 - TMC DSNB and CTL Updates (see page 427).

Job Control Statements

```
//stepname EXEC PGM=TMSUDSNB
//              [,PARM='[DATEFMT=(fmt)][,NOVER][,AUDIT][,TEST]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMC      DD  DSN=CAI.CA1.TMC,DISP=SHR
//AUDIT    DD  DSN=CAI.CA1.AUDIT,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//SYSIN    DD  *
(control statements here)
/*
```

Parameter Definition

PARM= specifies the functions which may be in effect during the processing. This parameter is optional and multiple values may be specified. Valid options are:

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 48. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

NOVER

Specifies that for Control Records, the record ID does not have to be verified. For DSNB records, the next DSNB pointer does not have to be verified. This parameter is optional.

AUDIT

Prints a hardcopy before and after image of TMC fields being updated. This parameter is optional.

TEST

No updates are performed. This parameter is optional.

More information:

TMSXTEND Utility—Dynamic TMC Extend Process Control Program (see page 472)

Control Statement Specification

TMSUDSNB uses four control statements: PASSWORD, CTLx/DSNB, VER, and REP. You can use more control statements for each program execution. The control statements are read through position 80. Sequence numbers within the statements cause errors.

PASSWORD Control Statement

The PASSWORD control statement is *required* to update a record. PASSWORD must be the *first* control statement in the SYSIN data set and only needs to be specified once. If only inquiry is performed, PASSWORD may be omitted. The format for the PASSWORD control statement is:

```
1        10
```

```
PASSWORD=xxxxxxxx
```

Control Statement Definition

PASSWORD

Is the control statement identifier and must start in position 1.

xxxxxxxx

Is the CA 1 system password.

CTLx/DSNB Control Statement

The CTLx/DSNB control statement identifies the type of record to inquire on or update. The format for the CTLx/DSNB control statement is:

```
1    6

CTLx [NOVER]
   or
DSNB nnnnnn [,NOVER]
```

Control Statement Definition

CTLx

Indicates Control Record #x is to be inquired on or updated. CTLx must start in position 1 and be followed by a space.

DSNB

Indicates a DSNB record is to be inquired on or updated. DSNB must start in position 1 and be followed by a space.

nnnnnn

Is the DSNB number.

NOVER

Indicates verification is not required before updating the associated record. If NOVER is omitted, verification is required.

VER Control Statement

The VER control statement verifies the value of a field before it is updated. It is required for the record ID (RECID) when updating Control Records. It is also required to specify the next record (NEXT) when updating DSNBs unless no verification (NOVER) is specified in the PARM of the EXEC statement or the CTLx/DSNB control statement. A Control Record or DSNB is not updated unless all VER and REP control statements for the record are correct. The format for the VER control statement is:

```
1   5

VER keyword=value,...,keyword=value
```

Control Statement Definition

VER

Is the control statement identifier. It must start in position 1 and be followed by a space.

keyword

Is one of the recognized TMSUDSNB keywords. For a list of recognized TMSUDSNB keywords, see the *Programming guide*.

value

Is the value to be verified. The value field may contain only the characters A-Z, 0-9, $, # or @. If the value field must contain any special characters or embedded blanks, the field must be enclosed in delimiters. Any special characters may be used as the delimiters (for example, 'C01 AC$TMS$Z' or *C01 AC$TMS$Z*). The value HEXZEROS indicates binary zeros.

REP Control Statement

The REP control statement indicates the new value of an updated field. A Control Record or DSNB is updated only if all VER and REP control statements for the record are correct. The format of the REP control statement is:

```
1   5

REP keyword=value,...,keyword=value
```

Control Statement Definition

REP

Is the control statement identifier. Start the identifier in position 1 and follow it by a space.

keyword

Is one of the recognized TMSUDSNB keywords. For a list of recognized TMSUDSNB keywords, see the *Programming Guide*.

value

Is the new value to be placed in the field that is specified by the keyword. The reserved word $REMOVE$, if used with the PASSWRD keyword, removes the password of a password-protected tape. The value field can contain only the characters A-Z, 0-9, $, # or @. If the value field must contain any special characters or embedded blanks, the field must be enclosed in delimiters. You can use any special characters as a delimiter (for example, 'C01 AC$TMS$Z' or *C01 AC$TMS$Z*). The keyword HEXZEROS can be used to replace a field with binary zeros. For date fields, binary zeros can be loaded using a keyword of ZEROS and the highest allowed value is 2155/366.

Examples:

1. Example 1 updates Control Record #2. The record ID is verified then USED is changed to 215.

   ```
   PASSWORD=system.password
   CTL2
   VER RECID=TMSCTL#2
   REP USED=215
   ```

2. Example 2 connects DSNB 675 to DSNB 680 after NEXT is verified.

   ```
   PASSWORD=system.password
   DSNB 675
   VER NEXT=0
   REP NEXT=680
   ```

3. Example 3 suppresses verification and makes the same update as Example 1.

   ```
   PASSWORD=system.password
   CTL2 NOVER
   REP USED=215
   ```

4. Example 4 shows a way for sites that have only a small test TMC/AUDIT used by 1 LPAR and a larger production TMC/AUDIT shared by multiple LPARs. Enable the Unique_RNAME feature to update directly control record 1 (TMSCTL#1) on the TMC/AUDIT used by only a single LPAR.

   ```
   PASSWORD=system.password
   CTL1
   VER RECID=TMSCTL#1
   REP FLAG1=40
   ```

Note: If multiple LPARs share the TMC, use TMSXTEND.

More information:

TMSXTEND Utility—Dynamic TMC Extend Process Control Program (see page 472)

TMS Report 48 - TMC DSNB and CTL Updates

This example shows successful updates to two DSNB records. NOVER was specified on the EXEC statement.

```
MONDAY, JUNE 20, 2005.177                        Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-48       TMC DSNB AND CTL UPDATES


         DSNB 104
         REP DSN=ADMIN.TEST.GNIKTAP
         REP VOLSER=000102
         REP EXPDT=YYYY/183

             DSNB 104    UPDATED.

         DSNB 1023
         REP DSN=ADMIN.TEST.NOSLEN
         REP VOLSER=000087
         REP EXPDT=PERM

             DSNB 1023   UPDATED.


                    02 TMC RECORDS UPDATED
```

TMSUDSNB Report Field Definitions

Control statement image

Input control statement

TMC RECORDS UPDATED

Number of DSNB and control records updated

TMC RECORDS NOT UPDATED - TEST MODE

Number of DSNB and control records that would have been updated if not in TEST mode

TMC UPDATES REJECTED

Number of DSNB and control records rejected due to errors

***** 'FLAG1' CANNOT BE CHANGED**

You attempted to turn on the Unique_RNAME indicator in CTL#1 but the TMC was not in the extended format. Unique_RNAME requires the TMC to be in extended format to be enabled.

TMSUNCAT Utility—Uncatalog Tape Data Sets

Use the TMSUNCAT utility to uncatalog tape data sets previously scratched by TMSCLEAN.

TMSUNCAT uses the EARLOUT data set created during TMSCLEAN SCRATCHLIST processing.

In environments where system catalogs containing tape data set entries are not available to the system where TMSCLEAN was executed, you can use TMSUNCAT to keep the system catalog and the TMC synchronized. TMSCLEAN uncatalogs what it can when scratching tapes based on the setting of the UNCATA option. The EARLOUT data set can be saved and used with TMSUNCAT on other systems to perform the tape data set uncatalog requests. See TMSCLEAN in this guide for more information.

CA 1 must be active or batch active to run TMSUNCAT.

Report Description

TMSUNCAT generates TMS Report 21 - Uncatalog Tape Data Sets Report. A sample of this report appears on TMS Report 21 - Uncatalog Tape Data Sets Report (see page 429). Optionally, TMSUNCAT also generates TMS Report 60 to list all errors that are found during processing. If no errors were found, the report only includes the phrase NO ERRORS WERE FOUND.

Job Control Statements

```
//stepname EXEC PGM=TMSUNCAT[,PARM='[,TEST]']
//STEPLIB DD DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT21 DD SYSOUT=A
//[TMSRPT60 DD SYSOUT=A]
//INDD DD DSN=scratch.list.earlout,
// DISP=OLD
```

Parameter Definitions

TEST

(Optional) Indicates that uncatalogs are not to be performed. The processed records are reported on the TMSRPT21 DD. If you specify the optional TMSRPT60 DD, the utility writes any errors that it finds to the report. Otherwise, the utility writes the statement NO ERRORS WERE FOUND.

Control Statement Specification

There are no control statements for TMSUNCAT.

TMS Report 21 - Uncatalog Tape Data Sets Report

```
MONDAY, NOVEMBER 25, 2013.329    Y O U R  C O M P A N Y  N A M E      P02049
PAGE=00001


TMS REPORT-21                                  TMSUNCAT TAPE DATA SETS REPORT

TEST parm received
CAI.UCT.TAPE2.FILE5                    on volume 001528 Bypassed because of
test parm
CAI.UCT.TAPE2.FILE4                    on volume 001528 Bypassed because of
test parm
CAI.UCT.TAPE2.FILE3                    on volume 001528 Bypassed because of
test parm
CAI.UCT.TAPE2.FILE2                    on volume 001528 Bypassed because of
test parm
CAI.UCT.TAPE2.FILE1                    on volume 001528 Bypassed because of
test parm
CAI.UCT.TEST.UNCAT                     on volume 001529 On Different Volser
CAI.UCT.TAPE6.FILE8                    on volume 001535 Bypassed because of
test parm
CAI.UCT.TAPE6.FILE7                    on volume 001535 Bypassed because of
test parm
CAI.UCT.TAPE6.FILE6                    on volume 001535 Bypassed because of
test parm
CAI.UCT.TAPE6.FILE5                    on volume 001533 Bypassed because of
test parm
CAI.UCT.TAPE6.FILE4                    on volume 001532 Bypassed because of
test parm
CAI.UCT.TAPE6.FILE3                    on volume 001532 Bypassed because of
test parm
CAI.UCT.TAPE6.FILE2                    on volume 001532 Bypassed because of
test parm
CAI.UCT.TAPE6.FILE1                    on volume 001531 Bypassed because of
test parm
CAI.UCT.TAPE6.FILE1                   on volume 001532 Skipped not volume 1
CAI.UCT.TAPE6.FILE1                   on volume 001533 Skipped not volume 1
CAI.UCT.TAPE6.FILE1                   on volume 001534 Skipped not volume 1
CAI.UCT.TAPE6.FILE1                   on volume 001535 Skipped not volume 1
CAI.UCT.TAPE6.FILE1                   on volume 001536 Skipped not volume 1
CAI.UCT.TAPE6.FILE1                   on volume 001537 Skipped not volume 1

Total Records Read                = 20
Total Volumes Read               = 9
Total Dsnbs   Read                  = 11
Total Errors Found                  = 0

Uncatalogs attempted           = 0
Uncatalogs successful            = 0
Bypassed not Volume 1         = 6
Bypassed because of TEST    = 13
```

TMSUNCAT Report Field Definitions

The report contains the following fields:

Total Records Read

Specifies the number of records that are read from the input file.

Total Volumes Read

Specifies the number of volume records that are read from the input file.

Total Dsnbs Read

Specifies the number of DSNB records that are read from the input file.

Total Errors Found

Specifies the number of errors that occurred during processing.

Uncatalogs attempted

Specifies the number of uncatalog requests that are processed.

Uncatalogs successful

Specifies the number of uncatalog requests that are processed successfully.

Bypassed not Volume 1

Specifies the number of UNCATALOG requests that are not processed because they are not on Volume 1.

Bypassed because of TEST

Specifies the number of UNCATALOG requests that are not processed because you specified the TEST parameter.

TMSUPDTE Utility—Apply Batch Updates to TMC Volume Records

Use the TMSUPDTE batch update utility to do the following:

- Update or verify any field in the TMC record or
- Apply mass updates to the TMC by selectively changing a field(s) within a range of volumes.

TMSUPDTE gives you the ability to update a single volume or a range of volumes *without affecting the scratch bit*. You can correct a bad internal VOLSER or assign EDMIDs for the entire TMC using a single set of VER/REP control statements. Using the *mass update* function of TMSUPDTE, you can select volumes for processing from a range of volumes to be processed based on the VER control statement, and identify the changes to be made using the REP control statements.

To resolve a NOT SCRATCH 28 error condition, you must reset the data set name on the Volume record. In most cases, the tape can be returned directly to the scratch pool because the status of the record has not been affected. TMSUPDTE does not allow a FLAG1 update to change the status of a tape to scratch or delete. For the scratch or delete bit to be on, it must have been set prior to the update.

CA 1 must be active or batch active to run TMSUPDTE.

Important! If the volumes being updated by TMSUPDTE need to be processed by TMSEXPDT to assign the proper retention, TMSEXPDT should be run prior to the TMSUPDTE step for the volumes records to be properly updated by TMSEXPDT. The control statement NOINTAL can also be used to indicate that the internal field was manually updated bit in FLAG1 should not be turned on.

Report Description

TMSUPDTE produces TMS Report 14 - TMC Batch Updates and TMS Report 14 - TMSUPDTE Error Cross-Reference.

The TMC Batch Updates report reflects the update, informs you of the update error, and gives warnings that the data set is expired. If a record is not found, you are notified. A sample of this report appears on TMS Report 14 - TMC Batch Updates (see page 440).

The Error Cross-Reference report provides a cross-reference to the update, giving the page number of the batch update report that contains the error and warning. A sample of this report appears on TMS Report 14 - TMSUPDTE Error Cross-Reference Report (see page 441).

Some volume record fields may be rejected for update, depending upon the 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 different, 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)
```

The following TMC fields cannot be updated by TMSUPDTE.

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

While the FLAG1 field can be updated, the scratch indicator (X'04) cannot be set by a batch update. The batch TMSCLEAN program and the CA 1 ISPF interface are the only means of marking tapes for scratch. Certain field updates related to multivolume data sets require that the update is performed to the first volume of the set (although this can be overridden with TMSUPDTE). These fields are:

```
ACCT
BLKSIZE
CDATE
CJOB
CPGM
CSTEP
CTIME
DEN
DSN
EXPDT
FLAG1
FLAG2
FLAG3
FLAG4
FLAG5
LABEL
LDATE
LJOB
LPGM
LRECL
LTIME
OUTCODE
OUTDATE
RECFM
SLOTS
SMSMC
TRTCH
1STDSNB
```

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. To bypass setting the internal field updated bit (X'80' in FLAG1), the NOINTAL keyword can be specified on the VOL control statement. The default fields to which this applies are:

```
EXPDT
FLAG1
FLAG2
FLAG3
FLAG4
FLAG5
LSTDSNB
NEXTVOL
NUMDSNB
PREVVOL
SLOT
VOLSER
VOLSEQ
1STDSNB
1STVOL
```

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

Update the following keywords with hexadecimal values.

These fields are:

```
CUNIT
DEN
LABEL
LUNIT
RECFM
ROBTY
TRTCH
```

Note: For more information about the values for each keyword, see the *CA 1 Tape Management Systems Programmer Guide*.

Job Control Statements

```
//stepname EXEC PGM=TMSUPDTE
//             [,PARM='[DATEFMT=(fmt)][,NODSN][,AUDIT][,TEST]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A
//SYSIN    DD  *
(control statements here)
/*
```

Parameter Definitions

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 14. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

NODSN

Specifies the data set name does not have to be verified. Normal processing requires the data set name to be verified prior to an update. This parameter is optional.

AUDIT

Prints the before and after image of the TMC fields being updated. This parameter is optional.

TEST

No updates are made to the TMC. This parameter is optional.

Important! CA recommends that you use the TEST parameter to verify your results before actually updating the TMC.

Control Statement Specification

There are six control statements for TMSUPDTE: VOL, VER, REP, TM, OI and NI. Multiple control statements are allowed for each program execution. The control statements are parsed through column 80. Do not use sequence numbers on these statements in columns 73 - 80.

VOL Control Statement

The VOL control statement identifies the TMC record(s) to be updated and is *required*. The format for the VOL control statement is:

```
1   5

VOL vvvvvv[-vvvvvv][,NODSN][,NOCHAIN][,SCRATCH]
                   [,UNEXPIRE][,NOINTAL]
```

Control Statement Definition

VOL

Is the control statement identifier. It must start in position 1 and be followed by a space.

vvvvvv

Is the volume serial number of the volume to be updated. If two volume serial numbers are indicated (separated by a dash), all volume serial numbers in this inclusive range are selected for update processing. If a range is requested, the NOCHAIN option *must* be specified. No volumes outside of the specified range are modified.

NODSN

Indicates the data set name does not have to be verified.

NOCHAIN

Indicates the volume to be updated is the second or higher volume sequence of a multivolume data set. Use this with caution because multivolume pointers may be affected. If NOCHAIN is omitted and the VOL control statement is for the first volume of a multivolume data set, updates to any of the listed fields are automatically made for all volumes of that multivolume data set. The fields are:

ACCTLDATE
BLKSIZE
CJOB or J
CDATE or C
CPGMLTIME
CSTEP
CTIME
DEN
DSN or D
EXPDT or E
LABEL
LJOB
LPGM
LRECL
OUTCODE or O
OUTDATE
RECFM
SMSMC
TRTCH

If NOCHAIN is specified, the updates are made only to the specified volume.

SCRATCH

Indicates that updates to the TMC Volume record should be made only if the volume is in scratch status.

UNEXPIRE

Indicates that the scratch bit is always turned off following an update. If this operand is omitted, the status of FLAG1 is ignored.

NOINTAL

Indicates that the volume being updated will not have the internal field updated indicator turned on (X'80' in FLAG1).

VER Control Statement

The VER control statement is used to verify the contents of a field before it is updated. VER is *required* for the data set name *unless* NODSN is specified either in the PARM of the EXEC statement or in the VOL control statement. The Volume record(s) is not updated unless all VER and REP control statements are correct. The format for the VER control statement is:

```
1   5

VER keyword=value,...,keyword=value
```

Control Statement Definition

VER

Is the control statement identifier. It must start in position 1 and must be followed by a space.

keyword

Is any keyword recognized by CA 1. TMC record definitions and keywords are detailed in the *Programming Guide*.

value

Is the value to be verified. The value field may contain only the characters A-Z, 0-9, $, # or @. If the value field must contain any special characters or embedded blanks, the field must be enclosed in delimiters. Any special characters may be used as the delimiters (for example, 'C01 AC$TMS$Z' or *C01 AC$TMS$Z*). The value HEXZEROS indicates binary zeros.

REP Control Statement

The REP control statement indicates the new value of an updated field. An asterisk (*) in position 1 indicates a comment and the statement is ignored. The Volume record(s) is not updated unless all VER and REP control statements are correct. The format for the REP control statement is:

```
1   5

REP keyword=value[,...,keyword=value]
```

Control Statement Definition

REP

Is the control statement identifier. It must start in position 1 and be followed by a space.

keyword

Is any keyword recognized by CA 1.

value

Is the new value to be placed in the field specified by the keyword. The reserved word $REMOVE$, if used with the PASSWRD keyword, will remove the password of a CA 1 password protected tape. The value field may contain only the characters A-Z, 0-9, $, # or @. If the value field must contain any special characters or embedded blanks, the field must be enclosed in delimiters. Any special characters may be used as the delimiters (for example, 'C01 AC$TMS$Z' or *C01 AC$TMS$Z*). The keyword HEXZEROS can be used to replace a field with binary zeros. For date fields, binary zeros can be loaded using a keyword of ZEROS and the highest allowed value is 2155/366.

Specify Julian dates in Julian format (yyddd or yyyy/ddd) or in your specified DATEFMT format. You *must* specify the CA 1 EXPDT keyword in one of the keyword formats defined in Specify CA 1 Keywords (see page 27).

Note: The yyddd format is not valid for the year 2000 and beyond.

Keyword dates are:

- 99000 CATALOG
- 90ddd CATLG/ddd
- 99365 PERM or PERMANENT
- 99366 PERM or PERMANENT
- 98ddd LDATE/ddd
- 99ccc CYCLE/ccc
- 88uuu USER/uuu

Note: To remove the password for a tape volume, you must also specify PASSWORD=xxxxxxxx on the VOL control statement where **xxxxxxxx** is the CA 1 system password.

TM Control Statement

The TM control statement is used to verify bit values in FLAG1, FLAG2, FLAG3 or FLAG4. This control statement is optional. If the TM control statement is coded, the TM value must be verified in the flag field for the volume record to be updated by the OI or NI instructions. The format for the TM statement is:

```
TM FLAGx=value
```

Control Statement Definition

TM

Is the control statement identifier. It must begin in position 1 and must be followed by a space.

FLAGx

Is any of the four flag fields in CA 1 volume records.

value

Is any valid bit setting in a flag field. At least one bit must be on; that is, '00' is not a valid value. All bit settings are detailed in the *Programming Guide*.

NI Control Statement

The NI control statement is used to turn off bit values in FLAG1, FLAG2, FLAG3 or FLAG4. If the TM control statement is coded, the TM value must be verified in the flag field for the volume record to be updated by the OI or NI instructions. The scratch bit (X'04' in FLAG1) **cannot** be changed by this operation. If you desire the scratch bit to be reset use the UNEXPIRE keyword on the Volume Control Statement. The format for the NI statement is:

```
NI FLAGx=value
```

Control Statement Definition

NI

Is the control statement identifier. It must begin in position 1 and must be followed by a space.

FLAGx

Is any of the four flag fields in CA 1 volume records.

value

Is any valid bit setting in a flag field. All bit settings are detailed in the *Programming Guide*.

OI Control Statement

The OI control statement is used to turn on bit values in FLAG1, FLAG2, FLAG3 or FLAG4. If the TM control statement is coded, the TM value must be verified in the flag field for the volume record to be updated by the OI or NI instructions. The scratch bit (X'04' in FLAG1) *cannot* be changed by this operation. The batch TMSCLEAN program is the only means of setting the scratch status. The format for the OI statement is:

```
OI FLAGx=value
```

Control Statement Definition

OI

Is the control statement identifier. It must begin in position 1 and must be followed by a space.

FLAGx

Is any of the four flag fields in CA 1 volume records.

value

Is any valid bit setting in a flag field. All bit settings are detailed in the *Programming Guide*.

Examples:

1. Example 1 creates the first volume of a multivolume data set, TMS.TEST.

```
VOL 100001
VER DSN=HEXZEROS
REP DSN=TMS.TEST,VOLSEQ=1
REP 1STVOL=100001,NEXTVOL=100002,PREVVOL=HEXZEROS
```

2. Example 2 creates the second volume of a two-volume data set, TMS.TEST.

```
VOL 100002,NOCHAIN
VER DSN=HEXZEROS
REP DSN=TMS.TEST,VOLSEQ=2
REP 1STVOL=100001,NEXTVOL=HEXZEROS,PREVVOL=100001
```

3. Example 3 verifies the data set name of the first volume (created in Example 1).

```
VOL 100001
VER DSN=TMS.TEST
```

4. Example 4 updates EXPDT and LJOB without verification.

```
VOL 100200,NODSN
REP EXPDT=ZEROS,LJOB=DFACCT01
```

5. Example 5 changes the vendor name for all volumes in the specified range but does not affect the status of any of these volumes. Only the VENDOR field and the UPDATED BY USER flag would be affected (and some Audit-related fields).

```
VOL 980001-981000,NOCHAIN,NODSN
REP VENDOR=IBM
```

6. Example 6 changes the EDMID from HSM to ASM2 for all volumes in the specified range. In this example, TMSUPDTE tests for HSM in the EDMID field and, if present, replaces HSM with the value ASM2. If HSM is not present, the volume is bypassed.

```
VOL 980001-981000,NOCHAIN,NODSN
VER EDMID=HSM
REP EDMID=ASM2
```

7. Example 7 verifies that the X'10' bit is turned on in FLAG1 and that the X'20' bit is on in FLAG3 before updating the volume to turn off the X'10' bit in FLAG1 and turn on the X'40' bit in FLAG1.

```
VOL 123456,NODSN
TM FLAG1=10
TM FLAG3=20
NI FLAG1=10
OI FLAG1=40
```

8. Example 8 changes the value of STATS/998 in the expiration date field to the value 97365 wherever found in the TMC.

```
VOL 000001-999999,NODSN,NOCHAIN
VER EXPDT=STATS/998
REP EXPDT=97365
```

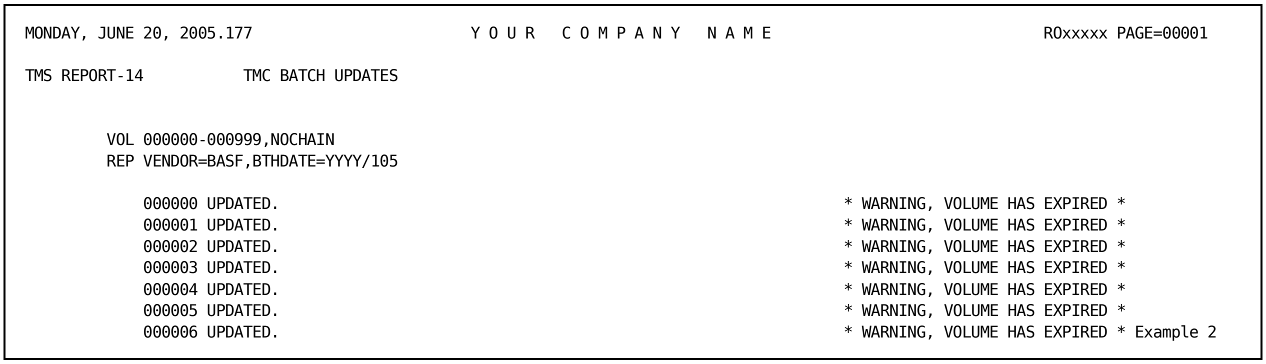
9. Example 9 changes the expiration date to a specific date for all tapes within a specific range that have an EDMID value of HSMO.

```
VOL H00001-H50000,NODSN,NOCHAIN
VER EMID=HSMO
REP EXPDT=YYYY/030
```

TMS Report 14 - TMC Batch Updates

This example shows a partial listing of the updates performed to the range of volumes 000000-000999, setting the VENDOR field to BASF and the BTHDATE field to YYYY/105. The second example shows output without the use of the AUDIT parameter. The third example (on the next page) shows output *with* the use of the AUDIT parameter.

Example 1



```
MONDAY, JUNE 20, 2005.177                    Y O U R   C O M P A N Y   N A M E                          ROxxxxx PAGE=00001

TMS REPORT-14           TMC BATCH UPDATES


         VOL 000000-000999,NOCHAIN
         REP VENDOR=BASF,BTHDATE=YYYY/105

             000000 UPDATED.                                                             * WARNING, VOLUME HAS EXPIRED *
             000001 UPDATED.                                                             * WARNING, VOLUME HAS EXPIRED *
             000002 UPDATED.                                                             * WARNING, VOLUME HAS EXPIRED *
             000003 UPDATED.                                                             * WARNING, VOLUME HAS EXPIRED *
             000004 UPDATED.                                                             * WARNING, VOLUME HAS EXPIRED *
             000005 UPDATED.                                                             * WARNING, VOLUME HAS EXPIRED *
             000006 UPDATED.                                                             * WARNING, VOLUME HAS EXPIRED * Example 2
```

```
WEDNESDAY, JUNE 15, 2005.179                 Y O U R   C O M P A N Y   N A M E                          ROxxxxx PAGE=00001

TMS REPORT-14          TMC BATCH UPDATES


         VOL 500092,SCRATCH,UNEXPIRE
         REP D=DASD.BKUP.A00100.C2
         REP DSN17=SD.BKUP.A00100.C2
         REP E=CYCLE/004
         REP V=1
         REP LABEL=02,RECFM=50,LRECL=32760,BLKSIZE=32760,BLKCNT=23870
         REP C=90175,CTIME=0127,J=SYSPBKUP,CSTEP=DASDBKUP,CDDNAME=A00100,CUNIT=500
         REP LDATE=90175,LTIME=0127,LJOB=SYSPBKUP,LUNIT=500
         *
             500092 UPDATED.

         VOL 500218,SCRATCH,UNEXPIRE
         REP D=DASD.BKUP.A00100.C2
         REP DSN17=SD.BKUP.A00100.C2
         REP E=CYCLE/004
         REP V=2
         REP LABEL=02,RECFM=50,LRECL=32760,BLKSIZE=32760,BLKCNT=23870
         REP C=90175,CTIME=0127,J=SYSPBKUP,CSTEP=DASDBKUP,CDDNAME=A00100,CUNIT=500
         REP LDATE=90175,LTIME=0127,LJOB=SYSPBKUP,LUNIT=500
         *
             500218 UPDATED.

         VOL 500068,SCRATCH,UNEXPIRE
         REP D=DASD.BKUP.A00100.C2
         REP DSN17=SD.BKUP.A00100.C2
         REP E=CYCLE/004
         REP V=3
         REP LABEL=02,RECFM=50,LRECL=32760,BLKSIZE=32760,BLKCNT=23870
         REP C=90175,CTIME=0127,J=SYSPBKUP,CSTEP=DASDBKUP,CDDNAME=A00100,CUNIT=500
         REP LDATE=90175,LTIME=0127,LJOB=SYSPBKUP,LUNIT=500
         *
             500068 UPDATED. Example 3
```

```
SUNDAY, JUNE 26, 2005.177                    Y O U R   C O M P A N Y   N A M E                          ROxxxxx PAGE=00001

TMS REPORT-14          TMC BATCH UPDATES


         VOL 000074
         VER DSN=ADMIN.TEST.BURMA
         REP E=90176

             EXPDT   CHANGED FROM - MMMDD YYYY
                             TO   - MMMDD YYYY
             000074 UPDATED.


                 01 TMC RECORDS UPDATED
```

TMS Report 14 - TMSUPDTE Error Cross-Reference Report

If any errors are encountered during TMSUPDTE processing, the page number containing the erroneous control statements are identified on this report. In the following example, errors were detected on pages 1, 2 and 5.

```
MONDAY, JUNE 20, 2005.178                    Y O U R   C O M P A N Y   N A M E                               PAGE=00007
```

```
TMS REPORT-14          TMSUPDTE - ERROR CROSS-REFERENCE REPORT

    ERRORS HAVE OCCURRED ON THE FOLLOWING PAGE NUMBERS:

       1     2     5
```

TMSUPDTE Report Field Definitions

Control statement image

Input control statement

TMC RECORDS UPDATED

Number of Volume records updated

TMC RECORDS WOULD HAVE BEEN UPDATED

Number of Volume records that would have been updated if not in TEST mode

TMC UPDATES REJECTED

Number of Volume record updates rejected

TMSVMEDT Utility—Validate the Vault Pattern Description Data Set

TMSVMEDT is one of several batch programs that comprise the CA 1 Vault Management System (VMS). These programs are executed in the following order:

1. TMSVMEDT reads the Vault Pattern Description Data Set, selects eligible TMC records, and creates work data sets for TMSVMVLT.
2. TMSVMVLT uses output from TMSVMEDT to determine movement into and out of off-site vaults and the tape library. This program creates a data set for use by TMSVMUPD and a report writer such as CA Earl.
3. TMSVMUPD uses output from TMSVMVLT to update the TMC with the new vaulting environment information.
4. Picking, distribution, and inventory reports can be created using the output data set from TMSVMVLT. A report writer such as CA Earl can produce the reports in any sequence.

CA Earl sample member TMEVLT01 is provided for the picking list, TMEVLT02 is provided for the distribution list, and TMEVLT03 is provided for the inventory list.

The *Administration Guide* provides a comprehensive discussion of the VMS.

The TMSVMEDT utility executes first in the VMS cycle and processes the Vault Pattern Description data set (VPD), validating and cross-referencing your site-defined movement data. Next, the TMC records associated with VMS control are selected and placed in a work file. TMSVMEDT produces TMS Report 23 - Vault Pattern Data Set, which describes the VPD and any errors it may contain. A sample of this report appears on page 462. Complete instructions for coding a VPD are included in the following discussion, *Defining the Vault Pattern Description Data Set*.

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, if a second execution of VMS is done later the same day, those additional volumes will be picked up and will join the first volume(s) at the correct off-site location.

Once the movement specified by the VPD has been completed, the tape(s) are returned to the library automatically.

CA 1 must be active or batch active to process TMSVMEDT.

User Exit

User exit TMSXVLT1 is provided for tailoring the TMSVMEDT TMC record selection processing. It is invoked during the TMC selection phase. It is called only if the option UXVLT1 is set to the name of the exit itself in the TMOOPTxx member of *hlq*.CTAPOPTN.

Note: For more information about the TMSXVLT1 user exit, see the *Programming Guide*.

Defining the Vault Pattern Description Data Set (VPD)

The VPD is a physical sequential 80 LRECL data set. It may be a single sequential file or an 80 LRECL member of a PDS.

Four control statements describe the off-site locations for VMS control, the data sets to be moved, and the desired movement. In addition, comments may be placed within the VPD to document the pattern content. The control statements and comment statements must begin in position 1.

- The Vault Name control statement defines location IDs (vaults) for use by VMS. The OUTCODE keywords (blank)LIB and LIBR do not need to be specified. The maximum number of slots for each vault may be defined.
- The Data Set Name control statement identifies the data sets to be controlled by VMS. Explicit data set names or CA 1 pattern masking specifications can be used.

- The Vault Pattern control statement specifies the location ID (vault) and period of retention for each data set under VMS control.
- The VOLUME or Cataloged Data Set Name control statements designate specific volume or volume sets that need to be moved to an off-site location, without regard for rules.
- The comment statement allows you to insert comments documenting your vault pattern. An asterisk (*) in position 1 is used to designate a comment, and the text for a comment may appear anywhere between positions 2 and 80 inclusive.

Control Statement Specification

Vault Name Control Statement

The Vault Name control statement defines an unlimited number of location IDs (vaults) for use by VMS. It must be the first control statement specified in the VPD and is *required*. Each location ID is from one to four characters inclusive. If the number of locations you want to define exceeds 72 positions, use multiple statements; a statement should not be split across two lines. If the number of slots in the vault needs to be defined, the vault name should be followed by the MAX= keyword. If the number of tapes assigned to a vault exceeds the value in the MAX= statement, a warning message is issued by TMSVMVLT. If a vault contains both reels and cartridges, the value specified for MAX= should be the larger value and not the total number of slots required. MAX= is not allowed for the LIB and LIBR vaults. The format for the Vault Name control statement is:

```
1      8

VAULTS=vlt1[,vlt2,MAX=nnnnnn,vlt3,...vltnn]
```

Example

```
VAULTS=SAFE,DRA1,MAX=1000,DRA2,PERM,MAX=500
```

In this example four vault names are assigned: SAFE, DRA1, DRA2 and PERM. Vault DRA1 has a slot capacity of 1,000 and PERM contains 500 volumes. The statement must begin on position 1, as shown.

VMS supports an unlimited number of user-defined locations. VMS automatically accommodates the locations (blank)LIB and LIBR, so these locations do not need to be defined on the VAULTS= statement.

The (blank)LIB and LIBR locations are commonly used for prestaging or delaying movement of a tape from the library while keeping the tape under VMS protection. The (blank)LIB and LIBR locations are listed on the reports. Locations (blank)LIB and LIBR are in-house, and the volumes are placed in their normal slot in the tape library. Tapes vaulted to LIBR may not be recreated, but tapes vaulted to (blank)LIB are eligible for recreation. Tapes marked to either location are considered protected by VMS until VMS relinquishes control. The MAX= statement cannot be specified for LIB and LIBR.

Each location defined on the VAULTS= statement supports an unlimited number of slots for tape reel media and the same number for cartridge media. VMS recognizes the media type and moves it to the appropriate sub-vault for the location ID. Reports generated reflect reel and cartridge sub-vaults. Slots may be limited by specifying the MAX= statement after the vault name definition.

Note: The first tape to initially enter a vault always begins with SLOT 1. VMS automatically vaults to the first empty slot for subsequent executions.

Data Set Name Control Statement

The Data Set Name control statement identifies explicit or pattern masked data set names to be controlled by VMS. This control statement is *required* and must begin in position 1 with DSN=ddd. The data set name (ddd) may be from 1 to 44 characters. If the data set name contains special characters (other than masking characters), it must be enclosed within single quotes, such as 'ABC$'. If any of the special characters in the data set name match a masking character, the character must be immediately preceded by a back slash (\). If you are specifying the entire data set name for a GDG do not put a "-" at the end to mask the "GnnnnVnnn" portion on the n ame. Reference the *Programming Guide* for complete details. Valid pattern masking characters do not need to be enclosed in single quotes. Any valid pattern mask may be specified for the data set name. You can specify any number of DSN= statements on a Vault Pattern control statement describing the desired movement.

You can optionally use keywords to reduce the number of entries in the VPD and change the movement criteria. For Generation Data Groups (GDGs), *do not* specify the generation/version number or relative generation because VMS does not consider the generation number during processing. If you wish to treat ABARs created tapes the same as GDGs, the system option ADGDG should be set to YES. This option allows the pseudo GDGs (CnnVnnnn) to be treated the same as regular GDGs (GnnnnVnn). The format for the Data Set Name control statement is:

```
1   5

DSN=ddd[,keyword,...,keyword]
```

Keyword Definitions

ABEND

Select data set even if the Closed by Abend indicator is on.

JOB=jjj

A complete, qualifying creating job name. The job name may be from one to eight characters and *must* be the complete job name, or pattern masking may be used. If the job name contains special characters (#, @, $, and so on), do not enclose it in delimeters. Only data sets created by job jjj are to be vaulted.

SEPDSN

Vault separately by complete data set name the tapes that meet index criteria. This keyword is only meaningful when the DSN value is a high-level character string (ddd-). This does not apply to unique generations of a GDG under the same DSN.

SEPJOB

Vault by creating job name. Multiple jobs that create the same data set will have a separate retention based on the job name. This keyword cannot be used with JOB=jjj.

REDEFINED

VMS must recalculate the vaulting requirements for the data set because the requirements have changed. Remove the REDEFINED keyword once the new vaulting environment is established. If the REDEFINED keyword is left on a statement once redefinition has occurred, it creates extra overhead in VMS processing time and can result in unnecessary reslotting of volumes.

Note: When using REDEFINED to change vault location IDs, be sure to have the old and new vault IDs in the VAULTS= statement. If the old ID is removed prior to the REDEFINED, the tapes in that location are ignored by the TMSVMEDT.

BACKWARD

Reverse movement through the vaults if earlier cycles are permanently removed from vault storage. Remove the BACKWARD keyword once the cycles are reset.

Examples:

DSN=ABC

DSN ABC is controlled by VMS.

DSN=ABC,JOB=jjj

DSN ABC, if created by job jjj, is controlled by VMS.

DSN=ABC-

ABC are the leading characters of the matching DSN, and all tapes with this match are controlled by VMS.

DSN=ABC-,SEPDSN

Each DSN under the ABC- character string is controlled separately. SEPDSN is meaningful only when the DSN value is a high-level character string.

DSN=ABC,SEPJOB

Each creating job name for DSN ABC is controlled separately. If both JOB= and SEPJOB are specified, SEPJOB is ignored. SEPJOB can be used with SEPDSN.

DSN=ABC,...,REDEFINED

The vault pattern is being changed and should be recalculated with the next VMS execution. Remove this option once the change is established.

DSN=ABC,...,BACKWARD

A tape has been prematurely and permanently removed from a vault, and the pattern should be recalculated to fill in the empty slot. Remove this option once the change is established.

More information:

Job Control Statements (see page 461)

VOLUME or Cataloged Data Set Name Control Statements

The VOLUME (VOL=) and Cataloged Data Set Name (CDSN=) control statements designate specific volumes or volume sets that are to be moved to an off-site location without regard to the rules specified in Data Set Name control statements (DSN=).

```
1   5
```

```
VOL=vvvvvv[-vvvvvv,keyword=value]
```

```
1    6
```

```
CDSN=ddd
```

Keyword Definitions

RTS=xxx

Specifies when to select stacking volumes for vault movement. **IMMED** is the default. If you specify **RTS=IMMED** or if you omit **RTS=**, you specify the volumes for vault movement when one or more active data sets are found. If xxx is **FULL**, you select stack volumes for vault movement only when the Realtime Stacking component of CA 1 flagged the volumes as full. You can specify the RTS= parameter only on the VOL= control statement. The parameter is ignored for nonstack volumes. The TMC volume record TMFLAG5 bit x'80' (keyword **STF**) indicates that the volume is a stack volume. TMFLAG4 bit x'01' (keyword **SNO**) indicates that the volume is full and no further stacking is to be done.

The VOLUME or Cataloged Data Set Name control statements differ from the Data Set Name control statements in three ways.

1. The VOLUME (VOL=) or Cataloged Data Set Name (CDSN=) control statements identify explicit volumes to be shipped to a single vault location. These explicit rules take precedence over Data Set Name (DSN=) control statements and are used in cases where a specific volume needs to be sent offsite. If the CDSN= control statement is used, a catalog lookup is performed and the statement is turned into a VOL= statement internally for the volume that is returned from the MVS catalog. If the VOL= statement (either explicitly stated or implied) is part of a multivolume chain, all volumes in the chain are moved together. If the CDSN= control statement is used, use an explicit data set name. For Generation Data Groups (GDGs), either specify the explicit GDG number, or the relative generation. The current generation is obtained from the catalog when TMSVMEDT is run, causing a new volser to be selected for vaulting. As a result, the previous VOLSER is returned from the vault. If the VOLSER requested is already vaulted, it is moved to the vault associated with the VOL= or CDSN= statements.

   Note: If you specify CDSN=, the specified data set name is not cataloged, and you do not specify the IGNERRS parameter, a U4000 abend occurs.

2. The RTS keyword is the only extra keyword that you can use with the VOLUME control statement. By definition, only the specified volumes are sent off-site or left off-site, regardless of the abend indicator or jobname. The Cataloged Data Set Name control statement does not support any keywords.

3. Movement is only to the first off-site location that the following Vault Pattern control statements specify. The volumes remain at that location until you remove the VOLUME or Cataloged Data Set Name control statement from the VPDD. When you remove the VOL= or CDSN= statement, the volser is moved back to the library. If the DSN on the tape matches another rule, the DSN is moved to that vault location.

Example:

```
VOL=123456                    Volume 123456 and all volumes chained
                              to it will be moved to the off-site
                              location that is specified in the
                              first Vault Pattern control statement
                              following this statement.

VOL=111122-111133             The volume range of 111122 through
                              111133 will be sent to the vault.

VOL=445000-446000,RTS=IMMED   The volumes in range 445000 through
           or                 446000 with one or more active data
VOL=445000-446000             sets are to be selected for movement
                              to the vault. Stack volumes with one
                              or more files are vaulted immediately,
                              that is, they do not have to be marked
                              full before vaulting.

VOL=200100-200200,RTS=FULL    Volumes in the range 200100 through
                              200200 with the stack bit on (TMFLAG5
                              x'80', keyword STF) will be selected
                              for vault movement only if the full bit
                              (TMFLAG4 bit x'01', keyword SNO) is
                              also on. Volumes in the range with the
                              stack bit off will be selected for vault
                              movement if the volume contains one
                              or more active data sets.

CDSN=PAYROLL.MASTER.G0018V00  The volume that generation 18 of the
                              PAYROLL.MASTER file resides on, and all
                              volumes chained to it, will be moved to
                              the off-site location that is specified
                              in the first Vault Pattern control
                              statement following this statement.
```

Notes:

- To specify a range, provide the starting volume, followed by a (-), followed by the ending volume. The presence of the dash (-) for use in the VOL statement indicates that a range is provided and does not represent an index or pattern masking character.
- If a volume range was specified, only active volumes in that range are sent off-site with *no* volume chaining being performed.

Vault Pattern Control Statement

The Vault Pattern control statement defines the location (vault name) where data sets are to be moved and the period of time they are to remain in the vault. This control statement is *required* and must begin in position 1. The vault name (vltx) must be either (blank)LIB LIBR, or previously defined by the Vault Name control statement. When you use (blank)LIB for the vault name, the blank character is not required (V=LIB).

If multiple Data Set Name control statements precede the Vault Pattern control statement, the movement described by the Vault Pattern control statement is applied to each of the previous Data Set Name control statements. A maximum of 28 Vault Pattern control statements may be specified for a given group of Data Set Name control statements.

In addition to the vault name (vltx), one or more of the keywords which follow must be specified. When two or more keywords are specified, they are treated as OR conditions. Two consecutive Vault Pattern control statements for the same vault are treated as AND conditions. See Vault Pattern Conditional Processing (see page 454) for examples. The format for the Vault Pattern control statement is:

```
1 3    8

V=vltx,keyword=value[,keyword=value,...,keyword=value]
```

Keyword Definitions

C=n

Vault for n cycles. Up to 255 cycles can be specified. A cycle is defined to VMS through a CA 1 system option. See Defining a Cycle to Vault Management (see page 459).

CR=n

Vault for n days from creation. Do not use with the E= keyword. Up to 9999 days can be specified.

VR=n

Vault for n days from OUTDATE. Do not use with the E= keyword. Up to 9999 days can be specified.

EXP

Vault until the data set is expired.

E={preferred date|Julian date|PERM}

Vault until specified date. The value of E= may be an actual date in your preferred date format, a Julian date (yyyy/ddd), or PERM for permanent vaulting. Whatever date format you choose, the date must be a date in the future.

LU=n

Vault until n days after data set was last used. Up to 9999 days can be specified.

If either VOL= or CDSN= control statements were used in the preceding group of control statements, those volumes are moved *only* to the first specified location. They will remain at that location until the VOL= or CDSN= control statement has been removed, regardless of the keyword(s) that were included on the Vault Pattern control statement.

Important!

If you specify a preferred date for E=, any subsequent revision to the DATEFMT option in *hlq*.CTAPOPTN or to the DATEFMT=(fmt) parameter in the TMSVMEDT program JCL may result in an abend if the date format is changed. Therefore, use *extreme caution* when revising either of these patterns.

You are strongly advised to change the date pattern of preferred date so that it reflects the newly revised date pattern specified in *hlq*.CTAPOPTN or in the TMSVMEDT program JCL. This restriction applies only to preferred date; if you are using a date in Julian format or the CA 1 keyword PERM, this warning can be ignored.

Examples:

V=VLT1,C=5

Retain a tape in VLT1 for five cycles or creations of the data set.

V=VLT2,CR=3

Retain a tape in VLT2 for three days from creation date (CDATE).

V=VLT3,VR=8

Retain a tape in VLT3 for eight days from the vault date (OUTDATE).

V=VLT4,E=2015/365

Retain a tape in VLT4 until the Julian date 2015/365.

V=VLT5,EXP

Retain a tape in VLT5 until the volume expires (EXPDT less than or equal to the VMS system run date).

V=VLT6,LU=5

Retain a tape in VLT6 for 5 days after its last use.

Comment Statement

Use an asterisk (*) in position 1 of the VPD to insert comment statements within the pattern. You may include comment text between positions 2 and 80 inclusive.

Vault Pattern Description Data Set Example

This example shows a Vault Pattern Description that clearly documents the data sets and desired VMS movement characteristics.

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* SAFE - IS LOCATED IN THE FIREPROOF STORAGE ROOM IN THE BASEMENT         *
* DRA1 - IS LOCATED OFF-SITE FOR DAILY/WEEKLY STORAGE                     *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
VAULTS=SAFE,DRA1
*
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* THE FOLLOWING STATEMENTS DESCRIBE MOVEMENT OF DAILY SAFE BACKUP TAPES   *
* AND SUBSEQUENT MOVEMENT TO DRA1 BEFORE RETURN TO LIBRARY.  EACH VOLUME  *
* IS STORED IN SAFE 1 CYCLE, THEN MOVED TO DRA1 FOR 1 CYCLE, THEN MOVED   *
* TO THE LIBRARY UNDER VMS CONTROL IN LIBR UNTIL IT EXPIRES.              *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
DSN=PAYROLL.DAILY-
DSN=SMF.BACKUP-,SEPDSN
DSN=MASTER.FILE.DAILY-
V=SAFE,C=1
V=DRA1,C=1
V=LIBR,EXP
*
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* THE FOLLOWING STATEMENTS DESCRIBE MOVEMENT OF WEEKLY DATA SETS.  DASD   *
* ARCHIVES, SYSRES BACKUPS, WEEKLY MASTER AND WEEKLY PAYROLL FILES ARE    *
* SENT TO DRA1 FOR 1 CYCLE AND THEN RETURNED TO THE LIBRARY.              *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
DSN=DASD.ARCHIVE-,SEPDSN
DSN=SYSRES.BACKUP-
DSN=MASTER.FILE.WEEKLY-
DSN=PAYROLL.WEEKLY-
V=DRA1,C=1
*** END OF PATTERN ***
```

Vault Pattern Processing

VMS Default Control

VMS default control is enforced when either of the following conditions exists:

- A tape has been assigned an OUTCODE equal to a vault name (including (blank)LIB and LIBR) and no entry exists for the DSN in the VPD.
- A DSN entry is removed from the VPD without checking in the volumes under VMS control for the DSN in question.

Once volumes are expired, VMS indicates that you should return the volumes to the library. An exception appeared in the TMSVMVLT output, and the Picking and Distribution Lists show the movement.

VMS Exceptions

The TMSVMVLT program reports on exceptions encountered during processing. Patterns that had no movement and duplicate slot numbers are examples of exceptions that would be noted on output of this program.

Avoid using the following keyword combinations when defining your patterns:

- JOB= and SEPJOB should not be used on the same DSN= statement. JOB= will be honored.
- CR= and E= cannot be used on the same V= statement.
- VR= and E= cannot be used on the same V= statement.
- LU= and E= cannot be used on the same V= statement.
- If V=vltx is specified more than once in a pattern, the locations should be contiguous.

Example:

Invalid	**Valid**	**Valid**
DSN=ddd	DSN=ddd	DSN=ddd
V=VLT1,VR=5	V=VLT1,VR=5	V=LIB,CR=2
V=VLT2,C=1	V=VLT2,C=1	V=VLT1,C=5
V=VLT1,EXP	V=VLT3,EXP	V=VLT1,VR=1

The example on the left is invalid because VLT1 is specified as both the first and last location on a pattern. This has been corrected in the example in the middle. The example on the right illustrates multiple specifications for VLT1.

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 as well as the Vantage/GUI display.

These exported virtual volumes do not appear on picking or distribution list (since there is no physical cartridge to move). They are displayed on the inventory list along 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 Pattern Conditional Processing

Vaulting with OR Conditions

If a tape is to be moved from a vault based on more than one criterion, OR processing logic can be used:

```
DSN=ABC
V=VLT1,C=3,EXP
```

This pattern directs VMS to hold data set ABC in VLT1 for either three cycles (C=3) or until the volume expires (EXP). Only one of the criteria must be met for VMS to move the volume to the next location. In the example above, the volume is returned to the library.

Vaulting with AND Conditions

To hold a tape in multiple locations or the same location using AND conditions, the logic can be specified as:

```
DSN=ABC
V=VLT1,C=3
V=VLT1,EXP
```

With this pattern, VMS holds the volume in VLT1 until it expires (C=EXP), and retains it in VLT1 for three cycles (C=3). Both criterion must be met for VMS to move the volume to the next location. In the example above, the volume is returned to the library from VLT1.

Note: Both AND and OR processing logic can be used in the same pattern:

```
DSN=ABC
V=LIB,C=1
V=VLT1,C=3,CR=40
V=VLT1,EXP
```

Vault Pattern Examples

Vault Pattern Example 1 - Vaulting By Data Set Name

The pattern is defined as follows:

```
VAULTS=SAFE
DSN=PAYROLL.DAILY1
DSN=PAYROLL.DAILY2
DSN=PAYROLL.DAILY3
V=SAFE,C=1
```

The data sets processed are:

VOLSER	Data Set Name	OUTCODE
000001	PAYROLL.DAILY1	SAFE
000002	PAYROLL.DAILY2	SAFE
000003	PAYROLL.DAILY3	SAFE

This example illustrates VMS action when the pattern describes an explicit data set name. In this instance, each data set moves to the SAFE vault for 1 cycle. The most recent creations of PAYROLL.DAILY1, PAYROLL.DAILY2 and PAYROLL.DAILY3 are stored in the SAFE location.

Vault Pattern Example 2 - Vaulting by Data Set Name with Index (-)

The pattern is defined as follows:

```
VAULTS=SAFE
DSN=PAYROLL.DAILY-
V=SAFE,C=1
```

The data sets processed are:

VOLSER	Data Set Name	CDATE	OUTCODE
000001	PAYROLL.DAILY1	95050	----

VOLSER	Data Set Name	CDATE	OUTCODE
000002	PAYROLL.DAILY2	95051	----
000003	PAYROLL.DAILY3	95052	SAFE

The Index (-) option allows VMS to control all data sets using the same initial character string. In this example, only the most recent creation of any DSN beginning with PAYROLL.DAILY is moved to SAFE. VOLSER 000003 is the most recent creation (95052).

Vault Pattern Example 3 - Vaulting by Data Set Name with Index (-) and SEPDSN

The pattern is defined as follows:

```
VAULTS=SAFE
DSN=PAYROLL.DAILY-,SEPDSN
V=SAFE,C=1
```

The data sets processed are:

VOLSER	Data Set Name	CDATE	OUTCODE
000001	PAYROLL.DAILY1	95050	SAFE
000002	PAYROLL.DAILY2	95051	SAFE
000003	PAYROLL.DAILY3	95052	SAFE

Because SEPDSN is specified, VMS considers the explicit DSN under the character string and applies the movement to each DSN. This is the same movement as when the VPD read:

```
VAULTS=SAFE
DSN=PAYROLL.DAILY1
DSN=PAYROLL.DAILY2
DSN=PAYROLL.DAILY3
V=SAFE,C=1
```

Vault Pattern Example 4 - Vaulting by Data Set Name with SEPJOB

The pattern is defined as follows:

```
VAULTS=SAFE
DSN=PAYROLL.DATA,SEPJOB
V=SAFE,C=1
```

The data sets processed are:

VOLSER	Data Set Name	CDATE	CJOB	OUTCODE
000001	PAYROLL.DATA	95050	PDALY	----
000002	PAYROLL.DATA	95051	PDALY	SAFE
000003	PAYROLL.DATA	95045	PWKLY	----
000004	PAYROLL.DATA	95052	PWKLY	SAFE

When more than one job can create a vaultable data set, SEPJOB may be used to qualify the vaulting by creating job name. In this example, jobs PDALY and PWKLY create the data set PAYROLL.DATA; VMS maintains separate cycles for each.

This is the same as coding:

```
VAULTS=SAFE
DSN=PAYROLL.DATA,JOB=PDALY
DSN=PAYROLL.DATA,JOB=PWKLY
V=SAFE,C=1
```

Vault Pattern Example 5 - REDEFINED Vault Patterns

The REDEFINED keyword instructs VMS to recalculate the requirements for the specified DSN. Any change to a pattern requires the use of REDEFINED for the next VMS execution. Once the new pattern is established, REDEFINED should be removed. If the redefinition involves the use of a new vault name, the VAULTS= statement should reflect both the old and new locations, or errors will occur.

Note: Use of the REDEFINED keyword allows volumes previously released from VMS control using the '(blank)VMS' outcode to be reconsidered for vaulting.

The following example shows the REDEFINED action on the number of cycles to be stored in a vault, and a vault name change. Once the new environment has been established, the REDEFINED keyword is removed from the Data Set Name control statement.

Normal Run	Tapes Vaulted VOLSER	Data Set Name	OUTCODE
VAULTS=SAFE,DRA1	000001	XYZ	SAFE
DSN=XYZ	000002	XYZ	DRA1
V=SAFE,C=1	000003	XYZ	DRA1
V=DRA1,C=2			

Normal Run	Tapes Vaulted VOLSER	Data Set Name	OUTCODE
VAULTS=SAFE,DRA1	000001	XYZ	SAFE
DSN=XYZ,REDEFINED	000002	XYZ	SAFE
V=SAFE,C=2	000003	XYZ	DRA1
V=DRA1,C=1			

Vault Pattern Example 6 - BACKWARD Vault Patterns

The BACKWARD option is used when a tape has been prematurely removed from a vault and is not to be returned.

The following example shows the action of BACKWARD when the SAFE vault had its oldest cycle of XYZ removed. The version of the data set in the DRA1 vault was sent back to SAFE. Once the BACKWARD cycle has performed the revaulting, BACKWARD should be removed from the DSN statement.

Normal Run	Tapes Vaulted VOLSER	Data Set Name	OUTCODE
VAULTS=SAFE,DRA1	000001	XYZ	SAFE
DSN=XYZ	000002	XYZ	removed
V=SAFE,C=2	000003	XYZ	DRA1
V=DRA1,C=1			

Normal Run	Tapes Vaulted VOLSER	Data Set Name	OUTCODE
VAULTS=SAFE,DRA1	000001	XYZ	SAFE
DSN=XYZ,BACKWARD	000003	XYZ	SAFE
V=SAFE,C=2	000002		moved back
V=DRA1,C=1			

Vault Pattern Example 7 - VOL=, CDSN=, and DSN= Patterns

The VOL= and CDSN= statements take precedence over DSN= patterns even if specified in the same group of patterns.

```
VAULTS=VLT1,VLT2
DSN=PAYROLL.MASTER
VOL=123456
CDSN=PAYROLL.MASTER.G0018V00
V=VLT1,C=1
V=VLT2,C=2
```

Volser	Data Set Name	Vault	Outcode EXPDT
123456	PAYROLL.MASTER.G0001V00	VLT1	96030
000011	PAYROLL.MASTER.G0015V00	----	CATALOG
000523	PAYROLL.MASTER.G0016V00	----	CATALOG
000353	PAYROLL.MASTER.G0017V00	VLT2	CATALOG
000565	PAYROLL.MASTER.G0018V00	VLT1	CATALOG
000343	PAYROLL.MASTER.G0019V00	VLT2	CATALOG
000533	PAYROLL.MASTER.G0020V00	VLT1	CATALOG

Even though generation 1 is no longer cataloged, it resides at location VLT1 as long as the VOL=123456 remains. Volume 000565 will remain at location VLT1 while generation 18 of the PAYROLL.MASTER file is cataloged to it.

Defining a Cycle to Vault Management

The CA 1 System Options Table defines a cycle to VMS through the CYD parameter in member TMOOPTxx of *hlq*.CTAPOPTN. This option can be set to YES; the default is NO.

The default value (NO) for CYD should be used if each creation of a data set, regardless of its creation date, is to be treated as an exclusive cycle.

```
VAULTS=SAFE
DSN=PAYROLL.DATA
V=SAFE,C=1
```

Current Data Sets	VOLSER	CDATE	CTIME	Current OUTCODE
PAYROLL.DATA	000001	95050	0830	----
PAYROLL.DATA	000002	95050	1715	----

Current Data Sets	VOLSER	CDATE	CTIME	Current OUTCODE
PAYROLL.DATA	000003	95051	0830	----
PAYROLL.DATA	000004	95051	1715	SAFE

Volume 000004 has the most recent CDATE and CTIME, so it is vaulted.

The TMOOPTxx CYD parameter should be set to YES if all creations of a given data set that are created on the same date are to be considered collectively a single cycle.

```
VAULTS=SAFE
DSN=PAYROLL.DATA
V=SAFE,C=1
```

Current Data Sets	VOLSER	CDATE	CTIME	Current OUTCODE
PAYROLL.DATA	000001	95050	0830	----
PAYROLL.DATA	000002	95050	1715	----
PAYROLL.DATA	000003	95051	0830	SAFE
PAYROLL.DATA	000004	95051	1715	SAFE

Since both volumes 000003 and 000004 were created on 94051, the CYD option set to YES instructs VMS to move both volumes.

Report Description

TMS Report 23 lists the VPD read by TMSVMEDT, and then provides a cross reference showing the pattern number assigned and the options requested for each. A sample of this report appears on TMS Report 23 - Vault Pattern Data Set (see page 462).

Job Control Statements

```
//TMSVMEDT EXEC PGM=TMSVMEDT
//             [,PARM='[DATEFMT=(fmt)][,IGNERRS][,TEST]']
//STEPLIB  DD  DSN=CAI.CTAPLINK,DISP=SHR
//TMSRPT   DD  SYSOUT=A                          <-- TMS REPORT 23
//VMSVPD   DD  DSN=&.&VMSVPD.,                     <-- OUTPUT FOR     *NOTES
//             DISP=(NEW,PASS,DELETE),               TMSVMVLT
//             UNIT=SYSDA,
//             SPACE=(CYL,(10,2)),
//             DCB=BLKSIZE=2640
//VMSTMC   DD  UNIT=SYSDA,
//             SPACE=(CYL,(10,2))
//VMSTMC2  DD  DSN=&.&VMSTMC2.,                    <-- OUTPUT FOR     *NOTES
//             DISP=(NEW,PASS,DELETE),               TMSVMVLT
//             UNIT=SYSDA,
//             SPACE=(CYL,(10,2)),
//             DCB=BLKSIZE=3960
//SYSOUT   DD  SYSOUT=A
//SYSPRINT DD  SYSOUT=A
//SORTLIB  DD  DSN=SYS1.SORTLIB,
//             DISP=SHR
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK06 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SYSIN    DD  *
(control statements here)
/*
//
```

Notes:

- The maximum record length for this file is 2636, variable block.
- The LRECL for this file is 396, fixed block.

Parameter Definitions

Three optional PARM values are available for use with TMSVMEDT:

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 23. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

Note: If a date pattern is defined and then subsequently revised, it may be necessary to reflect the pattern change in the VPD wherever the optional keyword, E=preferred date, appears in a Vault Pattern control statement. Read the caution text on Vault Pattern Control Statement (see page 450) and then revise the format of preferred date as necessary.

IGNERRS

Specifies error handling processes:

- Vault name errors will result in U4000-0006 ABEND.
- Duplicate data set names will result in a condition code of 8. The pattern associated with the first occurrence of the data set name is used.
- Any other errors will result in a condition code of 4.

TEST

Validates the VPD only. The output data sets VMSVPD, VMSTMC and VMSTMC2 are not created.

Completion Codes

When executed without any PARM values, TMSVMEDT will abend with a U4000-0006 if any error exists in the VPD. In a production mode, you are advised not to use any PARM values except DATEFMT.

TMS Report 23 - Vault Pattern Data Set

The first part of TMS Report 23 lists the VPD read by TMSVMEDT. TMSVMEDT flags any invalid entries with an asterisk (*) and prints a descriptive message to the right of the entry. The message identifies the nature of the error and provides other relevant information. A list of vault pattern error messages follows.

The second part shows the pattern number assigned by TMSVMEDT to each pattern, the job name qualifier (if used by the pattern), and the options specified. The record sequence number for the Data Set Name control statement can also be listed.

```
SUNDAY, FEBRUARY 27, 2005.057                    Y O U R   C O M P A N Y   N A M E                                    PAGE=00001

TMS REPORT-23                ***   VAULT PATTERN DATA SET   **

PARMS RECEIVED: TEST

              VAULTS=SAFE,DRA1
              DSN=DASD.BKUP.AO0100.C2
              DSN=DASD.BKUP.AO0101.C2
              DSN=DASD.BKUP.AO0102.C2
              DSN=DASD.BKUP.AO0103.C2
              DSN=DASD.BKUP.AO0104.C2
              DSN=DASD.BKUP.AO0105.C2
              DSN=DASD.BKUP.AO0106.C2
              DSN=DASD.BKUP.AO0107.C2
              DSN=DASD.BKUP.AO0108.C2
              DSN=DASD.BKUP.AO0200.C2
              DSN=DASD.BKUP.AO0201.C2
              DSN=DASD.BKUP.AO0202.C2
              DSN=DASD.BKUP.AO0203.C2
              DSN=DASD.BKUP.AO0204.C2
              DSN=DASD.BKUP.AO0205.C2
              DSN=DASD.BKUP.AO0206.C2
              DSN=DASD.BKUP.AO0207.C2
              DSN=DASD.BKUP.AO0208.C2
              V=LIBR,VR=1
              V=DRA1,EXP
              DSN=ADMIN.PAYR.DLY-
              DSN=ADMIN.PAYR.WLY-
              DSN=ADMIN.PAYR.MLY-
              DSN=ADMIN.PAYR.QLY-
              V=SAFE,CR=30
              V=DRA1,E=PERM,EXP
              DSN=SYSPROG.PROD-,SEPDSN
              DSN=ASM2-,SEPDSN
              V=LIBR,C=1
              V=DRA1,C=3,EXP
```

```
SUNDAY, FEBRUARY 27, 2005.057                    Y O U R   C O M P A N Y   N A M E                                    PAGE=00002

TMS REPORT-23                **   VAULT PATTERN DATA SET   **
PATTERN        DATA SET NAME                                JOBNAME   OPTIONS        INPUT
NUMBER                                                     QUALIFIER REQUESTED       CC 73-80

000001         DASD.BKUP.AO0100.C2
000002         DASD.BKUP.AO0101.C2
000003         DASD.BKUP.AO0102.C2
000004         DASD.BKUP.AO0103.C2
000005         DASD.BKUP.AO0104.C2
000006         DASD.BKUP.AO0105.C2
000007         DASD.BKUP.AO0106.C2
000008         DASD.BKUP.AO0107.C2
000009         DASD.BKUP.AO0108.C2
000010         DASD.BKUP.AO0200.C2
000011         DASD.BKUP.AO0201.C2
000012         DASD.BKUP.AO0202.C2
000013         DASD.BKUP.AO0203.C2
000014         DASD.BKUP.AO0204.C2
000015         DASD.BKUP.AO0205.C2
000016         DASD.BKUP.AO0206.C2
000017         DASD.BKUP.AO0207.C2
000018         DASD.BKUP.AO0208.C2
000019         ADMIN.PAYR.DLY-
000020         ADMIN.PAYR.MLY-
000021         ADMIN.PAYR.QLY-
000022         ADMIN.PAYR.WLY-
000023         ASM2-                                                 SD
000024         SYSPROG.PROD-                                         SD

        OPTIONS: SD=SEPDSN, SJ=SEPJOB, B=BACKWARD, A=ABEND, R=REDEFINED
               24 PATTERNS ASSIGNED"
```

TMSVMEDT Report Field Definitions

PARMS RECEIVED

Parameters passed to program

Control statement image

Input control statements

PATTERN NUMBER

Pattern number for VPD entry

DATA SET NAME

Data set name for VPD entry

JOBNAME QUALIFIER

Job name qualifier for VPD entry

OPTIONS REQUESTED

Other options selected for VPD entry

INPUT CC 73-80

User information for VPD entry

OPTIONS

List the available options for VPD entries

PATTERNS ASSIGNED

Number of VPD entries processed

TMSVMUPD Utility—Update the TMC for Vault Management

TMSVMUPD is one of three batch programs that comprise the CA 1 Vault Management System (VMS). These programs are executed in the following order:

1. TMSVMEDT reads the Vault Pattern Description Data Set, selects eligible TMC records, and creates work data sets for TMSVMVLT.
2. TMSVMVLT uses output from TMSVMEDT to determine movement into and out of off-site vaults and the tape library. This program creates a data set for use by TMSVMUPD and a report writer such as CA Earl.
3. TMSVMUPD uses output from TMSVMVLT to update the TMC with the new vaulting environment information.
4. Picking, distribution, and inventory reports can be created using the output data set from TMSVMVLT. A report writer such as CA Earl can produce the reports in any sequence.

The *Administration Guide* provides a comprehensive discussion of the VMS.

Using the data set created by TMSVMVLT, TMSVMUPD updates the TMC to reflect the new vault pattern configuration. The data set created by TMSVMVLT is compared against the TMC. If a change in the vaulting fields is present in the input, the update is performed. All updates appear on TMS Report 26.

CA 1 must be active or batch active to process TMSVMUPD.

Report Description

TMS Report 26 - CA 1 Vault Update Listing shows each TMC record updated by TMSVMUPD. A sample of this report appears on TMS Report 26 - CA 1 Vault Update Listing (see page 467).

Job Control Statements

```
//TMSVMUPD EXEC PGM=TMSVMUPD,
//             [PARM='[DATEFMT=(fmt)][,TEST]'],
//             COND=(4,LT,TMSVMVLT)             <-- DON'T RUN IF PROBLEMS
//STEPLIB  DD  DSN=CAI.CTAPLINK,
//             DISP=SHR
//TMSRPT   DD  SYSOUT=A
//VMSTAPE  DD  DSN=CAI.CA1.VAULT,               <-- FROM TMSVMVLT
//             DISP=OLD
```

Parameter Definitions

Two optional parameters are available for use with TMSVMUPD:

DATEFMT=(fmt)

Specifies the date format to be printed on TMS Report 26. The date pattern indicated by (fmt) is limited to 10 bytes and must be enclosed in parentheses. This parameter is optional. For more information on valid date formats, see Overriding the Preferred Date Pattern (see page 21).

TEST

Is used only when testing the Vault Management System. When PARM=TEST is specified, it suppresses actual updates to the TMC but still generates TMS Report 26 to show the updates that would have taken place. This parameter is optional.

Note: When TEST is used, do not move the tapes.

Completion Codes

Possible completion codes and their meanings are listed below.

0

Successful completion

12

Unable to update a TMC record. This condition code is issued, for example, if the data set name in the TMC record is different from that in the vaulting record created by TMSVMVLT. Information about the error is written to TMS Report 26. Processing continues for all other updates.

Important! If you receive a Completion Code of 12, at least one update has not been made for a volume that will be listed in the picking, distribution, and inventory reports.

TMS Report 26 - CA 1 Vault Update Listing

```
MONDAY, JUNE 20, 2005.177                        Y O U R   C O M P A N Y   N A M E                             ROxxxxx PAGE=00001

TMS REPORT-26                CA-1 VAULT UPDATE LISTINGS

VOLUME                                                 UPDATED TMC RECORD            PREVIOUS
SERIAL          DATA SET NAME                          VAULT SLOT    DATE       VAULT SLOT    DATE

500029  ASM2.BKUP.MASTER                              DRA1   002  MMMDD YYYY           000
500075  SYSPROG.PROD.SMFXA2.BKUP                      DRA1   007  MMMDD YYYY           000
500120  SYSPROG.PROD.SMFXA1.BKUP                      DRA1   014  MMMDD YYYY           000
500008  DASD.BKUP.A00107.C2                           LIBR   000  MMMDD YYYY           000
500026  ASM2.ARCH.MASTER                              LIBR   000  MMMDD YYYY           000
500027  ASM2.BKUP.MASTER                              LIBR   000  MMMDD YYYY           000
500050  DASD.BKUP.A00106.C2                           LIBR   000  MMMDD YYYY           000
500068  DASD.BKUP.A00100.C2                           LIBR   000  MMMDD YYYY           000
500073  SYSPROG.PROD.SMFXA2.BKUP                      LIBR   000  MMMDD YYYY           000
500080  SYSPROG.PROD.MVSXA1.BKUP                      LIBR   000  MMMDD YYYY           000
500082  DASD.BKUP.A00103.C2                           LIBR   000  MMMDD YYYY           000
500090  SYSPROG.PROD.MVSXA2.BKUP                      LIBR   000  MMMDD YYYY           000
500092  DASD.BKUP.A00100.C2                           LIBR   000  MMMDD YYYY           000
500114  SYSPROG.PROD.SMFXA1.BKUP                      LIBR   000  MMMDD YYYY           000
500154  DASD.BKUP.A00101.C2                           LIBR   000  MMMDD YYYY           000
500204  DASD.BKUP.A00106.C2                           LIBR   000  MMMDD YYYY           000
500218  DASD.BKUP.A00100.C2                           LIBR   000  MMMDD YYYY           000
500309  DASD.BKUP.A00103.C2                           LIBR   000  MMMDD YYYY           000
500334  DASD.BKUP.A00102.C2                           LIBR   000  MMMDD YYYY           000
500378  DASD.BKUP.A00105.C2                           LIBR   000  MMMDD YYYY           000
500782  DASD.BKUP.A00102.C2                           LIBR   000  MMMDD YYYY           000
500910  DASD.BKUP.A00108.C2                           LIBR   000  MMMDD YYYY           000
500986  DASD.BKUP.A00104.C2                           LIBR   000  MMMDD YYYY           000
500998  DASD.BKUP.A00106.C2                           LIBR   000  MMMDD YYYY           000
501022  DASD.BKUP.A00102.C2                           LIBR   000  MMMDD YYYY           000
501135  DASD.BKUP.A00105.C2                           LIBR   000  MMMDD YYYY           000
501176  DASD.BKUP.A00101.C2                           LIBR   000  MMMDD YYYY           000
501264  DASD.BKUP.A00108.C2                           LIBR   000  MMMDD YYYY           000
501470  DASD.BKUP.A00107.C2                           LIBR   000  MMMDD YYYY           000
501828  DASD.BKUP.A00104.C2                           LIBR   000  MMMDD YYYY           000
501996  DASD.BKUP.A00108.C2                           LIBR   000  MMMDD YYYY           000
502008  DASD.BKUP.A00105.C2                           LIBR   000  MMMDD YYYY           000
502020  DASD.BKUP.A00101.C2                           LIBR   000  MMMDD YYYY           000
502580  DASD.BKUP.A00104.C2                           LIBR   000  MMMDD YYYY           000
502642  DASD.BKUP.A00105.C2                           LIBR   000  MMMDD YYYY           000
502856  DASD.BKUP.A00103.C2                           LIBR   000  MMMDD YYYY           000
502870  DASD.BKUP.A00107.C2                           LIBR   000  MMMDD YYYY           000
100001  ADMIN.PAYR.DLY.G0406V00                       SAFE   001  MMMDD YYYY           000
100050  ADMIN.PAYR.MLY.G0045V00                       SAFE   022  MMMDD YYYY           000
100073  ADMIN.PAYR.QLY.G0017V00                       SAFE   023  MMMDD YYYY           000
```

TMSVMUPD Report Field Definitions

VOLUME SERIAL

Volume serial number of tape volume

DATA SET NAME

Data set name of tape volume

UPDATED TMC RECORD

VAULT

New vault name of tape volume

SLOT

New slot number of tape volume

DATE

New out-of-area date of tape volume

PREVIOUS

VAULT

Old vault name of tape volume

SLOT

Old slot number of tape volume

DATE

Old out-of-area date of tape volume

Message

Processing message

TMC RECORDS UPDATED.

Number of Volume records updated

WOULD HAVE BEEN UPDATED.

Number of Volume records that would have been updated if not TEST mode

TMSVMVLT Utility—Process the Vault Pattern Description Data Set and TMC Records

TMSVMVLT is one of three batch programs that comprise the CA 1 Vault Management System (VMS). These programs are executed in the following order:

1. TMSVMEDT reads the Vault Pattern Description Data Set, selects eligible TMC records, and creates work data sets for TMSVMVLT.
2. TMSVMVLT uses output from TMSVMEDT to determine movement into and out of off-site vaults and the tape library. This program creates a data set for use by TMSVMUPD and a report writer such as CA Earl.

3. TMSVMUPD uses output from TMSVMVLT to update the TMC with the new vaulting environment information.
4. Picking, distribution, and inventory reports can be created using the output data set from TMSVMVLT. A report writer such as CA Earl can produce the reports in any sequence.

CA Earl sample member TMEVLT01 is provided for the picking list, TMEVLT02 for the distribution list, and TMEVLT03 for the inventory list. 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. Please see '**Vault name statements to define Vault Desc. in the Report Header**' within each of the CA Earl members for complete details. Make sure the vault name is entered in all caps.

The *Administration Guide* provides a comprehensive discussion of the VMS.

The VPD information and the selected TMC records are passed by TMSVMEDT to the TMSVMVLT program, which determines the volumes to be moved to and from each site. TMSVMVLT creates a data set describing this movement for the CA Earl program to use in reporting and for the TMSVMUPD program to perform TMC updating.

CA 1 must be active or batch active to process TMSVMVLT.

Report Description

TMSVMVLT generates TMS Report 24 - Exceptional Conditions Encountered During Processing to inform you of any problems or errors with the movement. A sample of this report appears on TMS Report 24 - Exceptional Conditions Encountered During Processing (see page 471).

Job Control Statements

```
//TMSVMVLT EXEC PGM=TMSVMVLT
//STEPLIB  DD  DSN=CAI.CTAPLINK,
//             DISP=SHR
//TMSRPT   DD  SYSOUT=A                          <-- TMS REPORT 24
//VMSVPD   DD  DSN=&.&VMSVPD.,                     <-- FROM TMSVMEDT
//             DISP=(OLD,DELETE)
//VMSTMC00 DD  DSN=&.&VMSTMC2.,                    <-- FROM TMSVMEDT
//             DISP=(OLD,DELETE)
//VMSOUT   DD  DSN=CAI.CA1.VAULT,                <-- OUTPUT FOR
//             DISP=(NEW,CATLG),                     TMSVMUPD AND
//             UNIT=TAPE,                            CA-EARL
//             LABEL=RETPD=30                                   *NOTES
//VMSTMC01 DD  UNIT=SYSDA,SPACE=(CYL,(4,4))                     *NOTES
//VMSTMC02 DD  UNIT=SYSDA,SPACE=(CYL,(4,4))                     *NOTES
//VMSTMC03 DD  UNIT=SYSDA,SPACE=(CYL,(4,4))                     *NOTES
//SYSOUT   DD  SYSOUT=A
//SYSPRINT DD  SYSOUT=A
//SORTLIB  DD  DSN=SYS1.SORTLIB,
//             DISP=SHR
//SORTWK01 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK02 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK03 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK04 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK05 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
//SORTWK06 DD  UNIT=SYSDA,SPACE=(CYL,(2,2),,CONTIG)
```

Notes:

- Keep the output data set created by the VMSOUT DD for a period of time because this data set provides the only means for reproducing the Picking and Distribution Reports. The LRECL for this file is 396, fixed block.
- The LRECL for these files is 396, fixed block.

Completion Codes

TMSVMVLT completion codes and their meanings are listed below.

0

No exceptions were encountered during processing.

4

A volume is removed from a vault because no pattern exists, or a pattern exists but the volume does not contain a vault name associated with the pattern and REDEFINED has not been specified. This code is also set if no volumes were processed for a given pattern or if the maximum number of slots for a vault are exceeded. This is the highest acceptable completion code.

8

Critical TMC fields were not initialized. See TMS Report 24 (see page 471) for the warning messages indicating which fields were not initialized.

12

Duplicate slot values were found in TMC records.

16

The vault pattern specified that the tape is to remain in the vault for a number of days from creation; however, the creation date of the tape is not valid.

or

The vault pattern specified that the tape is to remain in the vault until it expires; however, the expiration date of the tape is not valid or if the expiration date is LDATE/ddd, the LDATE is not valid.

or

the vault pattern specified that the tape is to remain in the vault for a number of days from the OUTDATE; however, the OUTDATE of the tape is not valid.

See the warning messages on TMS Report 24 (see page 471) for the fields which were not initialized.

TMS Report 24 - Exceptional Conditions Encountered During Processing

Any exceptions during TMSVMVLT processing are listed on TMS Report 24.

```
THURSDAY, JULY 21, 2005.201                      Y O U R   C O M P A N Y   N A M E                              ROxxxxx PAGE=00001

TMS REPORT-24               VAULT MANAGEMENT SYSTEM   *---EXCEPTIONAL CONDITIONS ENCOUNTERED DURING PROCESSING---*

PATTERN 0002 FOR DSN=ARCHIVE.RECORDS                              JOBNAME QUALIFIER=**NONE** -NO VOLUMES WERE PROCESSED
PATTERN 0004 FOR DSN=DASD.BKUP.A00208.C2                          JOBNAME QUALIFIER=**NONE** -NO VOLUMES WERE PROCESSED
PATTERN 0005 FOR DSN=TAPE.DETAIL.TOTALS                           JOBNAME QUALIFIER=**NONE** -NO VOLUMES WERE PROCESSED
PATTERN 0007 FOR DSN=SAVE.BKUP.ALTERS.X2                          JOBNAME QUALIFIER=**NONE** -NO VOLUMES WERE PROCESSED


         END OF REPORT 24  -  TOTAL ITEMS LISTED=        4
```

TMSVMVLT Report Field Definitions

IDENTIFIER

VPD entry pattern number and data set name

MESSAGE

Processing message

TMSXTEND Utility—Dynamic TMC Extend Process Control Program

Use the TMSXTEND utility to dynamically build a new TMC while tape processing is allowed to continue across the shared TMC environment.

You can add or remove volume ranges by specifying the desired ranges as SYSIN parameters. The TMSXTEND utility saves the volume ranges internally in a new TMC record type, the Volume Definition Record.

Note: For information about Volume Definition Records, see the *Programming Guide*.

You can also use TMSXTEND to assign a new unique RNAME to all enqueue or reserves of the TMC and AUDIT file. Unique RNAME is especially important when multiple TMCs exist within a single GRS/plex or MIM/plex environment. You can use the unique RNAME instead of TMS-TMC, TMS-AUD, and TMSAPEC. The unique RNAME includes the full DSN of the TMC and AUDIT files and the volser that the files reside upon.

To verify the shared TMC environment before you perform a switch, run TMSXTEND in TEST mode. TMSXTEND also provides a RECOVER function to return the TMC to its previous state in the event of a failed extend attempt. The utility is designed to check as many environmental details as possible before proceeding with a TMC switch, greatly reducing the likelihood of a problem.

TMSFORMT was previously used to define a new TMC or Audit and to extend an existing TMC. TMSXTEND now performs this functions, except for the formatting of a new Audit file. Customers with nonstandard volume serial numbers should continue to use TMSFORMT to extend the TMC or define a new TMC.

Restrictions for TMSXTEND

1. Nonstandard Volume Serial Numbers: A few CA 1 customers have defined nonstandard volume serial numbers in their TMC that TMSXTEND cannot process. A nonstandard VOLSER is defined as any VOLSER with alpha characters in the rightmost positions of the VOLSER or with blanks to the left of a 4- or 5- character VOLSER. Examples of nonstandard VOLSERs are 0920XT, 00010A, or '12345'. These customers must continue to run the special versions of CA 1 user exits TMSXITU and TMSXITE that they have coded to support these VOLSERs. Plan to eliminate the use of these VOLSERs over time to enable them to take advantage of the convenience of the TMSXTEND utility.

   Short 3, 4, or 5 character VOLSERs with the blank on the right are valid. And VOLSERs with alpha characters on the left or even middle are also valid. Examples of VALID VOLSERs would be '12345 ', 9XT456, and ABC123. For all these types of VOLSERs, you do not have to maintain the CA 1 TMSXITU and TMSXITE exits.

2. Reverse Conversion: **An IPL is necessary**. The output of the TMSXTEND utility is a new format TMC. The volume ranges defined internally in a new type of record, the Volume Definition record. A TMC created in this format is described as an Extended Format TMC. Once a TMC has been converted to this format, the only way to convert back to the previous TMC format is by running TMSSPLIT to SPLIT off (export) all TMC and DSNB entries to a sequential file. This process is followed by a MERGE process to import them into a TMC defined with TMSFORMT. This process can be time-consuming in larger installations. Re-activate user exits TMSXITU and TMSXITE with an IPL. Plan your conversion to the Extended Format TMC well and do not convert back.
3. User Written Programs: Customers who have written their own programs to read and/or update the TMC must check to ensure that they can process the new Extended Format TMC. In general, programs that do not process DSNBs should work without modification. Contact your local CA 1 Support organization if you need help in determining if your user written programs need modification.
4. CA VM:Tape: Customers running CA VM:Tape must apply maintenance to that product if it is configured to share a TMC with CA 1. If you are running CA VM:Tape and sharing a TMC, contact CA Technical Support for CA VM:Tape to obtain the necessary maintenance. The TMSAUDEX utility can be executed to determine if a VM:Tape system sharing the TMC is back level. Refer to the description of TMSAUDEX for further information.
5. The first time that you set the new Unique_RNAME parameter to YES in the SYSIN control statements, you cannot code any EXCLUDE control statements in the CPULIST control statements. Once Unique_RNAME has been activated, this restriction no longer applies. This restriction does not apply if the Unique_RNAME parameter is disabled.

Changes to existing utilities:

1. TMSREMOV should NOT be used to remove volume ranges already in DELETE status on an Extended Format TMC. Instead, the REMVOL control statement of TMSXTEND should be used to remove those volumes.
2. TMSFORMT should NOT be used when the input is an Extended Format TMC. If you need to add more volumes to a TMC that is already in the new format, run TMSXTEND again.
3. TMSBINQ reports report zero (0) as the number of ranges for a new format TMC. Since there is no longer a limit to the number of volume ranges, there is no reason to maintain a count field of them in the control records.
4. TMSBINQ is the online TMSTIQ display of the volume ranges. The CA 1 ISPF online display of the volume ranges are different on an Extended Format TMC. The display now shows the starting and ending VOLSER in each range and the starting and ending relative record number for each of those VOLSERs.

Report Description

Subprograms of TMSXTEND, TMSBLDVR and TMSAUDEX create two reports - TMS REPORT-80 and TMS REPORT-81 respectively.

Job Control Statements

```
//stepname EXEC PGM=TMSXTEND,REGION=0M,
// PARM='parm1,parm2 '
//TMSRPT80 DD SYSOUT=*   /* TMSBLDVR RPT */
//TMSRPT81 DD SYSOUT=*   /* TMSAUDEX RPT */
//SYSPRINT DD SYSOUT=*   /* IDCAMS OUTPUT */
//CAISNP DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSIN DD *
ADDVOL 001800-001899 ROBTY=02 VENDOR=BASF
ADDVOL VM5000-BM5099  /* adding a second range               */
REMVOL VT0100-VT0199  /* sample of removing a volume range   */
ADDDSNB 1000
/*
//CPULIST DD *
EXCLUDE CPU9  /* this is a test system we are removing */
INCLUDE CPU6  /* PROD system with no tape activity     */
/*
```

Parameter Definitions

The DIAG parameter can be specified alone or with the TEST or RECOVER parameter. Parameters may be specified in any order and separated by commas or blanks. If the RECOVER parameter is omitted the program will execute in extend mode.

DIAG

This parameter causes TMSXTEND to write diagnostic snap dumps to the CAISNP output. Snap dumps are created to display internal volume tables, key data areas and changes to control records.

RECOVER

This parameter causes TMSXTEND to run in recovery mode. In recovery mode TMSXTEND will determine the status of a failed extend. If the signal to switch to the new TMC has not yet been given to all CPU's sharing the TMC, it will rename and return the old TMC to its original status, updating any control records necessary. If the signal to switch to the new TMC has been given to all CPU's sharing the TMC, it will finish updating any control records necessary.

TEST

This parameter causes TMSXTEND to run in test mode. In test mode, TMSXTEND will perform all of the checks and most of the functions of the actual extend mode. The new TMC is created and synchronized but the TMCs are not renamed and the CPUs do not switch to the new TMC. This mode allows you to preview or test the extend process. In test mode, if errors are detected they will be reported but the process does not fail as it does with an actual extend.

JCL Considerations

Note: The JCL provided in this document assumes that the CA 1 load library is defined as a LINKLIST data set.

TMSRPT80

Defines the report output data set for the TMS REPORT-80. This DD statement is required.

TMSRPT81

Defines the report output data set for the TMS REPORT-81. This DD statement is optional.

SYSPRINT

Defines the report output data set used by IDCAMS, which is invoked to perform the TMC renames. This DD statement is required.

CAISNP

TMSXTEND writes snap dumps to this DD statement when requested by PARM=DIAG. This DD statement is required only if PARM=DIAG is specified.

SYSIN

Defines the input data set for volume range and DSNB control statements. This DD statement is optional. If you do not specify a SYSIN DD statement all volume ranges and DSNBs in the existing TMC will be copied to the new TMC. This approach may be used to copy a TMC to a new, blocked TMC without adding or removing any volume ranges.

CPULIST

Defines the input data set for CPU include or exclude control statements. This DD statement is optional and if not specified, all CPUs sharing the TMC must be found in the current Audit file.

Completion Codes

The following completion codes are issued by TMSXTEND:

0

Successful completion.

8

TMC extend failed. Refer to the TMS REPORT-80 or 81 reports and the job log to determine cause of failure.

U0006

Invalid control statement error. Check TMS REPORT-80 or 81 to determine cause of failure.

U0996

The current TMC could not be allocated. Check to insure that the TMC is cataloged on the system where you are running TMSXTEND.

U0997

The new TMC could not be allocated or opened. Refer to messages TMSVR03E or TMSVR05E in the chapter "Utility Report Messages" in the Message Reference Guide.

U1022

TMSXTEND had abended for one of many reasons.Refer to the abend codes in the section IEFTMS83 - SMF Abend in the chapter "Abend Codes" in the *Message Reference Guide*.

Procedures for Executing the TMSXTEND Utility

Planning

Planning is an important part of running the TMSXTEND utility, especially the first time. The TMSXTEND utility creates new Volume Definition records in the TMC. Every system sharing the TMC must be at the correct maintenance level to perform an extend. TMSXTEND checks the maintenance level of every system it finds having written records to the Audit file. Make sure that the output accurately reflects your shared TMC environment. If a system has not been active, with CA 1 tape activity being recorded to the Audit, that system is not known. If you start CA 1 later on that system and you do not apply the necessary maintenance, CA 1 fails to initialize. First run the TMSBLDVR utility in standalone mode and evaluate your TMC volume ranges. If you have coded CA 1 user exits TMSXITE and TMSXITU to support volume serial numbers with alpha characters in the rightmost position in the VOLSER, you are not able to use TMSXTEND.

Plan the Execution of TMSXTEND

To ensure seamless execution of TMSXTEND, verify the following prerequisites:

Follow these steps:

1. Before you first run TMSXTEND, upgrade every system that shares the TMC to r11.x or higher and IPL every system.
2. To verify that you have identified all of the systems updating the TMC as recorded in your active Audit file, run TMSAUDEX. Include any systems that you find in an old Audit and not in the active Audit explicitly with an INCLUDE control statement. If the system is not a production system, you can exclude it with an EXCLUDE statement. If the system should share the new TMC, upgrade it later.
3. Run the TMSBLDVR utility in standalone mode, specifying the ranges that you want to add or remove. TMSBLDVR analyzes the existing and new volume ranges and reports any possible errors. TMSBLDVR also reports on the amount of space that the new TMC requires to support the existing ranges and newly added ranges.
4. Fix the errors coming out of TMSBLDVR. If any volumes with alpha characters in the rightmost position are found, you are not able to use the TMSXTEND utility.
5. Verify that program TMSRINIT can be run on all systems in the shared TMC environment. Specify PARM=STATUS on the JCL.
6. If you are removing volume ranges from the TMC and you use the scratch subpooling feature of CA 1, ensure that any volume ranges being removed are not defined in a range statement in the TMOSCRnn member of CTAPOPTN.
7. To determine if the TMC is cataloged in any nonshared catalogs, review your catalog environment. Any recatalog actions that the TMSXTEND utility performs are performed from the catalog environment on the system where TMSXTEND runs.

More information:

Follow Up Procedures after Initial Run of TMSXTEND (see page 479)

Run TMSXTEND

To build dynamically a new TMC, execute TMSXTEND.

Follow these steps:

1. Allocate the new CA 1 TMC on the desired volume. Use the following naming convention. Use the current TMC name and add a suffix of '.N'. The TMSBLDVR report contains the space requirements of the new TMC.

   Example:

   Existing TMC name - SYS3.CA1.TMC

   New TMC name - SYS3.CA1.TMC.N

   Important! Catalog the new TMC name on every system that shares the TMC. TMSXTEND renames the production TMC to TMC.O and the TMC.N to TMC.

   Notes:

   - You can change the blocking factor for the TMC by allocating the new TMC with the desired blocksize. For more information about blocking the TMC, see the *Programming Guide*.
   - If more than 4,369 cylinders are required for the TMC, add DSNTYPE=LARGE to the DD statement in the allocation step for the TMC. TMSPTRS does not support DSNTYPE=LARGE. If DSNTYPE=LARGE is used, use TMSAPEC instead of TMSPTRS.

2. To ensure that any critical chaining errors in the DSNB free chain are corrected, run TMSPTRS or TMSAPEC. Minor errors in DSNB or volume records do not cause any problems for TMSXTEND (such as incorrect DSN17 fields or a bad 1STVSN field). Errors in the DSNB free chain can cause TMSXTEND to abend.

3. Plan to run TMSXTEND before or after the CA 1 daily maintenance or any other CA 1 batch job. Before you execute TMSXTEND, stop CA 1 batch utilities and shut down the DBS and APEC subtasks of CTS. Disable the CA Vantage™ Storage Resource Management script for CA 1. The ISPF interface and the old TSOTIQ interface work correctly for individual volume or DSNB inquiries or updates. The range-display in the ISPF interface does not work while TMSXTEND is in progress.

   Note: For more information about the DBS and APEC subtasks of CTS, see the *Administration Guide*.

4. Back up the TMC by running TMSCOPY. Perform this step within three hours of running TMSXTEND.

5. Run the TMSXTEND utility first with PARM=TEST. You verify that all systems are in the proper status and that the range/DSNB control statements are valid. This test also validates that the new TMC is large enough to hold all records that you requested.

6. Submit the TMSXTEND utility without the PARM=TEST and monitor its execution. The utility provides numerous messages indicating the status of the extend and the switch-over to the new TMC.

   Note: TMSXTEND issues the TMSXT09I SIGNAL ALL CPUS TO ACCESS THE NEW TMC message. Now run TMSRINIT on all CPUs that access this TMC. These TMSRINITs do not end immediately, they stay active until the final switch over command is issued (TMSXT21I). The TMSRINITs end within one minute after the TMSXT21I message is issued. Do not use PARM=STATUS on the TMSRINITs that are executed for this step. If TMSXTEND uses PARM=TEST, all messages are issued and you run TMSRINIT. The physical switch is not done. Set TMSRINIT as a started task.

7. Review messages that TMSXTEND issues. If TMSXTEND cannot automatically recover from abends, run TMSXTEND with PARM=RECOVER. To see if the automatic recovery already took place, refer to the messages that TMSXTEND issues.

8. Verify that all systems switched to the new TMC by checking the job log of the TMSXTEND utility. Review the system consoles for message TMSMR04I, which confirms the switch. To finalize the switch to the new TMC, TMSRINIT must run on all CPUs. Set TMSRINIT as a started task.

9. Create a postconversion backup of the TMC by running the TMSCOPY utility again.

   Note: If you execute TMSCOPY after a TMSXTEND, some records duplicate to the AUDUMP and AUDUMP2 files. Depending on your audit file size and the amount of processing, some records or records for several days can be duplicated. The duplicate records do not have any impact on a noncurrent restore.

10. If you stopped the DBS subtask or CTS in step 4, restart it now.

Follow Up Procedures after Initial Run of TMSXTEND

After you convert to an extended format TMC, retire exits U and E and verify the TMC.

Follow these steps:

1. Remove exits U and E from your LINKLIST library.

2. If you use a nonshared MVS/catalog environment, verify that the TMC is cataloged on the correct volume on all systems. For systems that are not in a shared catalog structure, uncatalog the old TMC. Catalog the TMC on the new volume where you allocated TMC.N.

Control Statement Specifications

Control

Control statements for the SYSIN DD statement

Note: The control statements are parsed thru column 80. Do not use sequence numbers on these statements in columns 73 - 80.

ADDDSNB nnnnnnn

Requests that nnnnnnn additional DSNB records be added to the new TMC.

ADDVOL

Lowvol-Highvol DEN=nn ROBTY=nn TRTCH=nn VENDOR=cccccccc DELETE specifies that the volume range lowvol-highvol be added to the new TMC. Optional parameters are provided to initialize the volume records with a density (DEN), robot type (ROBTY) recording technique (TRTCH) and vendor (VENDOR). For the DEN, ROBTY and TRTCH parameters, nn should be replaced with hexadecimal values. For a list of these values see the *Programming Guide*. The DELETE parameter simply indicates the volume(s) are added in DELETE status in the new TMC. VENDOR is an 1-8 character string that is used to identify the media manufacturer. In addition to a range, a single VOLSER may also be specified on the ADDVOL statement.

Notes:

- The dash ("-") is required between lowvol and highvol.
- VOLSERs shorter than six characters are supported. They are left justified and padded with blanks. The volumes added will be in SCRATCH status if you do not use the DELETE keyword in the control statement.

REMDSNB nnnnnnn

Requests that nnnnnnn free DSNB records be removed from the new TMC. Only DSNBs from the end of the free chain that have never been used can be removed.

REMVOL lowvol-highvol

Specifies that all volumes in the range lowvol-highvol be removed from the TMC. All volumes in this range must be in DELETE status, otherwise the entire range is not removed. A single VOLSER may also be specified on a REMVOL control statement.

UNIQUE_RNAME=[*YES*|*NO*]

Specifies if UNIQUE_RNAME is used with this TMC. If you want to code this statement, code it as the first statement in the SYSIN DD.

Control statements for the CPULIST DD statement

When all systems sharing the TMC have tape processing performed on them at least once a day, there is no need for any CPUs to be included in the CPULIST control statements. However, if there are some CPUs that have NO tape processing performed on them, or when the tape processing is very sporadic that can go for days at a time without any tape processing, it would be recommended to use the INCLUDE type of CPULIST control statement.

The EXCLUDE control statement should only be used when there is a CPU that shares the TMC AND that CPU will not be active when TMSXTEND is running AND that CPU is 100% guaranteed to be at the correct maintenance level applied. If that CPU will be active, but there will not be any tape jobs running on it, it should not be excluded using the EXCLUDE control statement.

You cannot exclude the CPU that TMSXTEND is running on itself. If the SMF ID of the CPU that TMSXTEND is running on is listed on the EXCLUDE statement, it will be automatically included and shown as forced on Report-81.

INCLUDE smf id

This control statement defines the SMF ID of a system running CA 1 that should be included in the list of systems to process when performing an extend function. INCLUDE must begin in column 1 followed by a blank and then the SMF id. Only one SMF id may be specified per control statement.

EXCLUDE smf id

This control statement defines the SMF ID of systems running CA 1 that should be excluded from processing by TMSXTEND. EXCLUDE must begin in column 1 followed by a blank and then the SMF id. Only one SMF id may be specified per control statement. Note that failure to find a system specified on an EXCLUDE statement is not considered an error.

Comments identified by an asterisk in column 1 may be specified in both CPULIST and SYSIN DDs but are not listed in any report.

TMS Report 80 - Report Field Definitions

```
TMS REPORT=80  JOB=TMSXTEND  STEP=EXTEND     THURSDAY, SEPTEMBER 26, 2013.269
TIME=16:38:10
THURSDAY, SEPTEMBER 26, 2013.269      Y O U R  C O M P A N Y  N A M E          P02138
PAGE=00001
     TMSBLDVR                                        VOLUME DEFINITIONS

  TMC BEING READ: EVAWI01.TEST.TMC
  TMC BEING CREATED: EVAWI01.TEST.TMC.N

      CURRENT TMC VOLUME RANGES
   EXTERNAL VSN     TMC REC NUMBER

  200000-200099    0000004-0000103
  8801  -8802      0000104-0000105
  DEL1  -DEL2      0000106-0000107
  DEL101-DEL200    0000108-0000207
  DEL250-DEL300    0000208-0000258
  DEL350-DEL400    0000259-0000309

  INPUT PARAMETERS

      UNIQUE_RNAME=YES
      ADDVOL DEL301-DEL349

  NUMBER OF VSNS  ALLOCATED IN CURRENT TMC: 00000421
  NUMBER OF DSNBS ALLOCATED IN CURRENT TMC: 00019000
  NUMBER OF FREE DSNBS:                     00019000
  NUMBER OF FREE DSNBS THAT CAN BE REMOVED: 00008999

  NEW TMC BEING CREATED: EVAWI01.TEST.TMC.N

  NEW TMC VOLUME RANGES

      200000-200099
      8801  -8802
      DEL1  -DEL2
      DEL101-DEL200
      DEL250-DEL300
  NEW DEL301-DEL349
      DEL350-DEL400

  TMSVR07I CREATE NEW VSN CONTROL RECORDS.
  TMSVR11I COPY/CREATE/REMOVE VOLUME RECORDS.
  TMSVR13I CREATE VOLUME DEFINITION RECORDS.
  TMSVR15I COPY/CREATE/REMOVE DSNB RECORDS.

  NUMBER OF VSNS  ALLOCATED IN NEW TMC: 00000470
  NUMBER OF DSNBS ALLOCATED IN NEW TMC: 00019000
  NUMBER OF FREE DSNBS IN NEW TMC:      00019000

  TMSBLDVR COMPLETED
```

Note: If you do not code the ADDDSNB or REMDSNB control statement, the number of DSNBs that can be deleted is not analyzed. The report displays the following line:

```
NUMBER OF FREE DSNBS THAT CAN BE REMOVED WAS NOT ANALYZED
```

External Volser Range

This field shows the low and high VOLSER numbers for a range found in the TMC.

TMC Rec Number

This field shows the low and high relative record number of the volume serial numbers for each range in the TMC.

TMS REPORT 80 also lists the TMC name being evaluated, the input control statements, and the ranges that would be created for the new TMC. The report also gives the number of DSNBs currently allocated, currently unused, and how many can be removed. If you run TMSBLDVR in standalone mode, the report also gives the space requirements of the TMC based on an unblocked TMC or a TMC allocated with a blocksize of 8840.

UNIQUE TMC RNAME ACTIVATED

Reason:

You are using use UNIQUE_RNAME=YES to create a TMC or you are extending an existing TMC.

Action:

None. The message is informational.

UNIQUE TMC RNAME DEACTIVATED

Reason:

You are using UNIQUE_RNAME=NO when you are extending an existing TMC and unique RNAME was previously activated.

Action:

None. The message is informational.

UNIQUE_RNAME= MUST BE FIRST CONTROL STATEMENT

Reason:

UNIQUE_RNAME is not the first control statement coded. TMSBLDVR ends with the reason code 12.

Action:

Code UNIQUE_RNAME as the first control statement.

TMSXTEND DETECTED AN ERROR DURING TMSBLDVR PROCESSING – TMSBLDVR IS ENDING

Reason:

An error was detected during TMSXTEND processing which caused TMSBLDVR to end prematurely. The new TMC has not been fully created.

Action:

Search and fix any errors in the TMSXTEND job and resubmit.

TMS Report 81 - Report Field Definitions

```
THURSDAY, APRIL 17, 2008.108         Y O U R   C O M P A N Y N A M E                                         ROxxxxx PAGE=00001
TMS REPORT-81                 AUDIT EXTRACT BY CPU
CPU  RECORD COUNT  LO--DATE   TIME  HI--DATE   TIME FLG CTL STMT ERRORS

CPU1  1,199        2008105    8:40   2008108   9:49  Y
CPU2    0          ZEROS      0:00   ZEROS     0:00  N  EXCLUDE
*********************************
 * CONTROL STATEMENTS PROCESSED:  *
*********************************

EXCLUDE CPU2
```

CPU

The SMF id of the z/OS system creating audit records. SMF ids for systems specified on an INCLUDE or EXCLUDE statement are also listed even if no audit records were found for the requested system. Users running VM Tape and sharing the TMC with that product may see a report line with a blank SMF id. This is an indication of the VM Tape system not being at the necessary maintenance level for sharing the TMC and is considered an error.

RECORD COUNT

Total number of audit records found for the system.

LO--DATE/TIME

Lowest date and time of any audit record found for the system.

HI--DATE/TIME

Highest date and time of any audit record found for the system.

FLG

Flag indicating if the system is at the required level to run the TMC extend utility TMSXTEND. All systems listed must be at the required level in order to run TMSXTEND. 'Y' or 'N' will be listed in this column indicating whether or not the system is at the right maintenance level.

CTL STMT

This column will show INCLUDE or EXCLUDE if a control statement was specified for this system.

ERRORS

This column provides a description of any errors encountered in processing the Audit file.

ERROR WRONG CODE LEVEL

This message will be issued by TMSAUDEX if TMSINIT has not run on that system or if there has been no tape processing since the last backup or the system does not have the required maintenance to support TMSXTEND. Either run TMSINIT or a tape job, or apply the correct maintenance to support TMSXTEND and rerun the procedure.

ERROR EXCLUDE NOT ALLOWED

Reason:

The Unique_RNAME feature was being activated when TMSAUDEX experienced an EXCLUDE statement. CPUs are not allowed to be excluded while the Unique_RNAME feature is being activated.

Action:

Cancel the Unique_RNAME activation by removing the "UNIQUE_RNAME=Y" or remove EXCLUDE control statements from TMSXTEND and resubmit.

TMSXTEND Recovery Procedures

If an abend situation occurs in TMSXTEND, it will probably recover by itself. Look for messages TMSXT41I and TMXST49I. If the machine that TMSXTEND is running on goes down, there will be some manual recovery necessary.

The first scenario is when the TMC being extended is shared by more than 1 CPU. This is the easiest situation to recover from. Simply run TMSXTEND with PARM=RECOVER on any other CPU that shares the same TMC and AUDIT.

The second scenario is when the TMC being extended is only accessed from 1 (one) CPU, and that CPU goes down. After it uses IPL and CAS9 has run, but BEFORE the TMSINIT job has been run, you may be required to perform some manual intervention. If the TMSXT30I message had been issued, but NOT the TMSXT31I message OR TMSXT21I messages, then you must manually rename TMC.O to TMC before running TMSINIT. In this situation, after TMSINIT has been run you should also run TMSXTEND with a PARM=RECOVER to finish the cleanup of CTL#4.

In the same scenario, if the TMSXT30I message had not yet been issued by TMSXTEND, then there is no recovery necessary and you may simply run TMSINIT.

Also in this same scenario, if the TMSXT31I AND TMSXT21I messages had both already been issued, there is also no recovery necessary and you may simply run TMSINIT.

Chapter 3: Producing Customized Reports Using CA Earl

This section contains the following topics:

About CA Earl

CA Earl, the CA Easy Access Report Language, provides you with the capability to design and produce customized reports. Easy access to TMC Volume and DSNB record information provides reporting flexibility and allows you to tailor reports to your desired format.

This chapter describes the types of access methods available for retrieving information from the TMC and identifies required CA Earl components. The discussion of date manipulation functions for date fields retrieved from the TMC includes coding examples of date function calls. CA Earl uses data record formats common to both CA 1 and CA TLMS Tape Management, so that access from either system is identical. The CA 1 CA Earl record layout identifies those fields that are unique to CA 1 and those that are unique to CA TLMS Tape Management. Several CA Earl coding examples are supplied, with the output generated.

Additional CA Earl information may be obtained from the *CA Earl Reference Guide*. A cross reference of supplied CA Earl members can be found in member EARLINDX of the CA 1 Source library.

Note: The support for TMSGRW is continued due to its wide usage. The current functionality of TMSGRW is stabilized while any new functionality is added to the CA Earl or CA Easytrieve interfaces to CA 1.

Accessing Information in CA 1

CA Earl provides different methods for accessing TMC Volume and DSNB record information. The desired access or *run type* (TYPRUN) is specified using a SET statement. The coding example on Using CA Earl (see page 501) illustrates the use of the SET statement.

The following access types are available:

SEQUENTIAL

TYPRUN = R_SEQUENTIAL access of the TMC returns all Volume and DSNB records in the TMC in their physical sequential order. This access type is most useful when a listing of all or selected data set are to be reported on. Volume records will be returned to CA Earl, one at a time, without any DSNB information. DSNB records will be passed to CA Earl without any Volume information. The retrieval process stops when the end of the TMC is reached. You must include a DD statement, with a ddname of TAPEDB pointing to the desired CA 1 TMC data set. CA 1 does not have to be active for this TYPRUN. This is the *default* access type.

VOLUME

TYPRUN = R_VOLUME access of the TMC returns all Volume records in the TMC in their physical sequential order. The retrieval process stops after all Volume records have been read. You must include a DD statement, with a ddname of TAPEDB, pointing to the desired CA 1 TMC data set. CA 1 does not have to be active for this TYPRUN.

CHAINED

TYPRUN = R_CHAINED access of the TMC reads Volume records sequentially and returns the information for each Volume record to CA Earl, a record at a time.

- If the Volume record indicates that DSNBs are chained to this volume, CA Earl reads the chain directly, combining volume and DSNB information in each record returned to CA Earl.
- If the Volume record indicates this volume is part of a multivolume chain, CA Earl reads the volume chain directly, looking for each volume with chained DSNBs. Each DSNB chain is then processed as described above.

The process terminates after all active records have been read. The TAPEDB DD statement is required for this access type and it *must* point to the production (active) TMC data set name. CA 1 must be *active or batch active* for this TYPRUN.

AUDIT

TYPRUN = R_AUDIT access is provided to return information obtained from the CA 1 Audit file. Data is provided for all CA 1 Audit record types. A complete description of the Audit record types and the information they contain is located in the discussion of the TMSAUDIT utility in this guide. Records are provided in their physical sequential order. You must include a DD statement, with a ddname of TAPEDB pointing to the desired CA 1 Audit file. See the sample members in the CA 1 Source library.

VAULT

TYPRUN = R_VAULT access is provided to process CA 1 vault information obtained from the CA 1 utility TMSVMVLT. Records are provided in their physical sequential order. You must include a DD statement, with a ddname of TAPEDB pointing to the data set created by the VMSOUT DD. CA 1 does not have to be active for this TYPRUN. For a sample CA Earl program to access the CA 1 vault information, see the sample members in the CA 1 Source library. Since several fields are provided which are not part of the EARLDEFS definition, you must use TMEYDEFV to access the data.

RETENTION

TYPRUN = R_RETENTION access is provided to process information obtained from the CA 1 utility programs TMSCLEAN, TMSCTLG, TMSCYCLE and TMSEXPDT. Records are provided in their physical sequential order. You must include a DD statement with a ddname of TAPEDB pointing to the output file from one of the above CA 1 utilities. CA 1 does not have to be active for this TYPRUN. Since several fields are provided which are not part of the EARLDEFS definition, you must use TMEYDEFR to access the data.

REPORT

TYPRUN = R_REPORT allows you to process information which was previously extracted from CA 1 using CA Earl with a TYPRUN value of VOLUME, SEQUENTIAL, CHAINED, or AUDIT. You can use the EARL source member TMERPT01 to create the extract records. The CA Earl PUT function is used to create the file. The field definitions are provided by the EARLDEFS member. You must include a DD statement with a ddname of TAPEDB, pointing to the output file from the extract run. You can use the EARL source member TMERPT02 to produce this output file from the extracted records created by TMERPT01. CA 1 does not have to be active for this TYPRUN.

CA Earl Components Description

The successful reporting of TMC information by CA Earl requires the following components:

- EARLDEFS Copy Member (see page 490)
- TMSEARL Interface (see page 490)
- CTSEDATE Date Subroutine (see page 490)

EARLDEFS Copy Member

A standardized CA Earl copy member, EARLDEFS, defines the fields in the CA 1 CA Earl record. This member is stored in the CA 1 Source library. A COPY statement is *required* to copy these field definitions into your CA Earl program.

TMSEARL Interface

The TMSEARL interface is called by CA Earl each time your CA Earl program issues a GET (explicit or implied).

TMSEARL interprets runtime parameters and internal switches and flags, and determines how to access the TMC. TMSEARL issues the appropriate read macro(s), formats an output record, and returns the output record to CA Earl.

CTSEDATE Date Subroutine

The manipulation of CA 1 internal dates is possible using the CTSEDATE date subroutine. This subroutine can be called from your CA Earl program.

CTSEDATE uses the CA 1 TMMDATE macro to provide the following functions:

- RETURN_DATE
- ADD_DAYS
- ADD_YEARS
- ADD_WORK
- NUM_DAYS
- TO_EXTERNAL
- SUB_DAYS
- SUB_YEARS

- FROM_EXTERNAL
- SET_FORMAT

Date Processing (see page 494) lists the date fields in the CA 1 CA Earl record that are available for manipulation. Coding examples and descriptions of the date functions begin on CTSEDATE Date Function Calls (see page 496). The date constants and variable fields are contained in the EARLDEFS copy member.

CTSEDATE passes the return code issued by TMMDATE to CA Earl for verification. CA Earl places the return code in a field equal to the subroutine's name, CTSEDATE. In nonzero return code situations, CTSEDATE always appends an error message to the CTSEDATE function code. This message contains the TMMDATE return code and reason code, and text which explains the failure. The text is identical to that defined for the TMMDATE Anchor control block described in the *Programming Guide*.

CA Earl can test the return code and verify an error condition (see example on Example 5 (see page 524).) Return codes issued by the TMMDATE macro are described in the *Systems Programming Guide*. The only exception is return code 32, which indicates that CA Earl issued a nonexistent function call.

CTSEPM Advanced Pattern Masking Subroutine

The CTSEPM subroutine provides advanced pattern masking support. You can call CTSEPM from your CA EARL program. CTSEPM uses the CTSPM module to provide the following functions:

- MATCH
- COMPARE
- VALIDATE
- CONVERT
- CONVERTX

CTSEPM passes the return code issued by CTSPM to CA EARL for verification. CA Earl places the return code in a field equal to the subroutine's name, CTSEPM. In non-zero return code situations, CTSEPM can return an optional error message.

CTSEPM Function Calls

The following examples illustrate the CTSEPM function calls. All calls use an implicit object class of MVSFILE, so the parameters are DSNs or DSN patterns.

The parameters vary by function, but the error message parameter is 24 characters long and is always optional. The FUNCTION parameter is 8 characters long. The data set parameter is 44 characters long and the pattern mask parameters are 88 characters long.

MATCH Function

Use the MATCH function to determine if a DSN matches a pattern mask.

A return code of zero indicates that the DSN matches the pattern.

A return code of four means the DSN does not match the pattern.

Higher return codes indicate an error.

Non-zero return codes are accompanied by a 24-character error message.

```
DEF PM_FUNC            (X 8) =  ' ' 'FUNCTION'
DEF PM_PAT1           (X 88) =  ' ' 'PATTERN MASK1'
DEF PM_MSG            (X 24) =  ' ' 'ERROR MESSAGE'
DEF PM_DSN            (X 44) =  ' ' 'DATA SET NAME'

SET PM_FUNC = 'MATCH'
SET PM_PAT1 = 'SYS#.DDA####.-'
SET PM_DSN = DSN
CALL CTSEPM USING PM_FUNC
                  PM_PAT1
                  PM_DSN
                  PM_MSG
```

COMPARE Function

Use the COMPARE function to compare two pattern masks and determine which one is more specific. You can use the function to arrange a pattern masks group in an order that allows your program to find the 'best fit' pattern mask.

Higher return codes indicate an error.

Non-zero return codes are accompanied by a 24-character error message.

A zero return code is accompanied by a message that matches the PM_MORE, PM_LESS, or PM_SAME values.

```
DEF PM_FUNC            (X 8) =  ' ' 'FUNCTION'
DEF PM_PAT1           (X 88) =  ' ' 'PATTERN MASK1'
DEF PM_PAT2           (X 88) =  ' ' 'PATTERN MASK2'
DEF PM_MSG            (X 24) =  ' ' 'ERROR MESSAGE'
DEF PM_MORE           (X 24) =  'MORE SPECIFIC           '
DEF PM_LESS           (X 24) =  'LESS SPECIFIC           '
DEF PM_SAME           (X 24) =  'SAME SPECIFIC           '
```

```
SET PM_FUNC = 'COMPARE'
SET PM_PAT1 = 'SYS#.DDA####.-'
SET PM_PAT2 = 'SYS#.-'
CALL CTSEPM USING PM_FUNC
                  PM_PAT1
                  PM_PAT2
                  PM_MSG

IF PM_MSG = PM_LESS
   SET FLG = 'Y'
ENDIF
```

VALIDATE Function

Use the VALIDATE function to determine if a pattern mask is valid.

Higher return codes indicate an error.

Non-zero return codes are accompanied by a 24-character error message.

```
DEF PM_FUNC            (X 8) =  ' ' 'FUNCTION'
DEF PM_PAT1           (X 88) =  ' ' 'PATTERN MASK1'
DEF PM_MSG            (X 24) =  ' ' 'ERROR MESSAGE'

SET PM_FUNC = 'VALIDATE'
SET PM_PAT1 = 'SYS#.DDA####.-'

CALL CTSEPM USING PM_FUNC
                  PM_PAT1
                  PM_MSG
```

CONVERT Function

Use the CONVERT function to convert a pattern mask into its internal format. Use the CONVERT function to improve the performance of later MATCH function calls. Internal format pattern masks can also be directly compared to determine which is more specific. This can replace a COMPARE function call.

A return code of zero indicates that the function successfully converted the pattern. The original pattern mask is replaced by its 44-character internal format.

Non-zero return codes are accompanied by a 24-character error message.

```
DEF PM_FUNC            (X 8) =   ' ' 'FUNCTION'
DEF PM_PAT1           (X 88) =   ' ' 'PATTERN MASK1'
DEF PM_MSG            (X 24) =   ' ' 'ERROR MESSAGE'

SET PM_FUNC = 'CONVERT'
SET PM_PAT1 = 'SYS#.DDA####.-'
CALL CTSEPM USING PM_FUNC
                  PM_PAT1
                  PM_MSG
```

CONVERTX Function

Use the CONVERTX function to convert a pattern mask from its internal format to its external format for display purposes.

If the return code is zero, then an 88-character external pattern mask replaces the original 44-character internal format.

Non-zero return codes are accompanied by a 24-character error message.

```
DEF PM_FUNC            (X 8) =   ' ' 'FUNCTION'
DEF PM_PAT1           (X 88) =   ' ' 'PATTERN MASK1'
DEF PM_MSG            (X 24) =   ' ' 'ERROR MESSAGE'
DEF INTERNAL_PAT      (X 44) =   ' ' 'INTERNAL PATTERN'

SET PM_FUNC = 'CONVERTX'
SET PM_PAT1 = INTERNAL_PATTERN
CALL CTSEPM USING PM_FUNC
                  PM_PAT1
                  PM_MSG
```

Date Processing

The CTSEDATE date subroutine can be called from your CA Earl program to provide date manipulation functions for date fields retrieved from the CA 1 TMC.

The following fields are stored in the CA 1 CA Earl record and may be used in date manipulations:

Common Area

EXPDT

The expiration date for the volume

P_EXP_TYPDAT

A field containing the CA 1 Version 5.1 expiration date type:

0—Zeros

4— Internal Julian date

8—Old Julian date format

12—FOREIGN (Non-CA 1 Control)

16—LDATE/ddd (Days Since Last Used Control)

20—CATALOG (Catalog Control)

24—CYCLE/ccc (Cycle Control)

28—PERMANENT (Permanent Retention Control)

32—CATLG/ddd (Catalog Days Control)

36—USER/uuu (User Control)

40—STATS/sss (Status Control)

Note: In many instances, you must know the type of expiration date for a volume to perform any meaningful date calculations based on EXPDT.

CDATE

Date that volume was created

LDATE

Date that volume was last used

DATECLN

Date that tape was last cleaned

BTHDATE

Date that tape was first used

MOVEDATE

Date that tape was moved out

RUNDATE

Current date

RUNTIME

Current time

CA 1 Specific Area

AUDATE

Date that volume was last updated.

CTSEDATE Date Function Calls

The following examples illustrate CTSEDATE date function calls. Each date function is defined as 80 bytes, which allows a message to be appended to the date function if the TMMDATE macro issues a nonzero return code.

Important! You must define seven packed decimal digits (P 7.0) for a four-byte packed decimal field.

RETURN_DATE Function—Return Current Date and Time

Use the RETURN_DATE function to return the current date and time. The date is a four-byte packed Julian date and the time is four-bytes.

```
DEF CUR_DAY      (P 7.0) = NONE 'TODAYS' 'DATE'
DEF CUR_TIME     (P 7.0) = NONE 'CURRENT' 'TIME'
DEF FUNCTION      (X 80) = NONE

SET FUNCTION = 'RETURN_DATE'
CALL CTSEDATE USING FUNCTION CUR_DAY CUR_TIME
```

ADD_DAYS Function—Add Days to a Date

Use the ADD_DAYS function to add a number of days to a given date, giving a resultant date. FROM_DATE is four-bytes, packed, CA 1 internal date; DAYS is four-bytes packed; and RESULT_DATE is four-bytes, packed, CA 1 internal date.

```
DEF FROM_DATE      (P 7.0) = NONE 'FROM' 'DATE'
DEF DAYS           (P 7.0) = NONE 'DAYS' 'ADDED'
DEF RESULT_DATE    (P 7.0) = NONE 'RESULT' 'DATE'
DEF FUNCTION        (X 80) = NONE

SET FUNCTION  = 'ADD_DAYS'
SET FROM_DATE = EXPDT
SET DAYS      = 7

CALL CTSEDATE USING FUNCTION
                    FROM_DATE
                    DAYS
                    RESULT_DATE
```

TO_EXTERNAL Function—Convert Internal Date to External Date

Use the TO_EXTERNAL function to convert a CA 1 internal date to a 10-byte character external date, according to the date format supplied.

```
DEF DATE_FMT     (X 10) = 'MMMDD YYYY'
DEF FUNCTION     (X 80) = NONE
DEF PRINT_DATE   (X 10) = NONE 'EXPDT' 'DATE'

SET FUNCTION = 'TO_EXTERNAL'
CALL CTSEDATE USING FUNCTION
                    PRINT_DATE
                    EXPDT
                    DATE_FMT
```

TO_EXPLODED Function—Convert Internal Date to Exploded Format

Use the TO_EXPLODED function to convert a CA 1 internal date to an exploded format. The extracted date related data is stored in the designated area. EARLDEFS provides an area to store the information as P_EXP_DATA. P_EXP_TYPDAT is a field name in the P_EXP_DATA area.

```
DEF FUNCTION     (X 10) = NONE
DEF P_EXP_DATA   (X 64) = NONE

SET FUNCTION = 'TO_EXPLODED'
CALL CTSEDATE USING FUNCTION
                    P_EXP_DATA
                    EXPDT
```

SET_FORMAT Function—Change External Date Format

Use the SET_FORMAT function to change the desired external date format. Remember that before you can issue the FROM_EXTERNAL date request, you must tell CTSEDATE the format of the external date. The SET_FORMAT function is provided for this purpose.

Additionally, you may use the SET_FORMAT function to set the desired output date format for internal CA 1 dates that you choose to print on CA Earl reports.

```
DEF FUNCTION     (X 80) = NONE
DEF DATE_FMT     (X 10) = NONE
```

```
SET FUNCTION = 'SET_FORMAT'
SET DATE_FMT = 'MMMDD YYYY'

CALL CTSEDATE USING FUNCTION
                    DATE_FMT
```

Note: When the CA 1 interface to CA Earl is first invoked, the SET is initialized to your installation DEFAULT setting. Use the SET_FORMAT function to change to your desired date format.

FROM_EXTERNAL Function—Convert CA Earl Date to CA 1 Date

Use the FROM_EXTERNAL function to convert an external date in your CA Earl program (for example, SET MYDATE = 'YYYY/MM/DD') to a CA 1 compatible internal date. You could use this date for comparisons, arithmetic operations, and so on, with other CA 1 internal date fields.

```
DEF MY_INT_DATE   (P 7.0) = NONE
DEF MY_EXT_DATE   (X 10)  = 'YYYY/MM/DD'
DEF DATE_FORMAT   (X 10)  = NONE

SET DATE_FORMAT = 'YYYY/MM/DD'
SET FUNCTION = 'SET_FORMAT'
CALL CTSEDATE USING FUNCTION
                    DATE_FORMAT

SET FUNCTION = 'FROM_EXTERNAL'
CALL CTSEDATE USING FUNCTION
                    MY_INT_DATE
                    MY_EXT_DATE
```

Note: Remember, before you can issue the FROM_EXTERNAL function, you must have called CTSEDATE with the SET_FORMAT function and the desired external date format (if different from your installation default date format).

ADD_YEARS Function—Add Years to a Date

Use the ADD_YEARS function to add a number of years to a date, producing a resultant date. All fields are four-bytes packed. Dates are CA 1 internal dates.

```
DEF FROM_DATE    (P 7.0) = NONE 'FROM' 'DATE'
DEF YEARS        (P 7.0) = NONE 'YEARS' 'ADDED'
DEF RESULT_DATE  (P 7.0) = NONE 'RESULT' 'DATE'
DEF FUNCTION      (X 80) = NONE
SET FROM_DATE = EXPDT
SET YEARS     = 3
SET FUNCTION  = 'ADD_YEARS'

CALL CTSEDATE USING FUNCTION
                    FROM_DATE
                    YEARS
                    RESULT_DATE
```

ADD_WORK Function—Add Work Days to a Date

Use the ADD_WORK function to add a number of work days to a date, giving the resultant date.

```
DEF FROM_DATE      (P 7.0) = NONE 'FROM' 'DATE'
DEF DAYS           (P 7.0) = NONE 'DAYS' 'ADDED'
DEF RESULT_DATE    (P 7.0) = NONE 'RESULT' 'DATE'
DEF FUNCTION        (X 80) = NONE
SET FUNCTION  = 'ADD_WORK'
SET FROM_DATE = EXPDT
SET DAYS      = 7

CALL CTSEDATE USING FUNCTION
                    FROM_DATE
                    DAYS
                    RESULT_DATE
```

NUM_DAYS Function—Calculate Days Between Dates

Use the NUM_DAYS function to calculate the number of days between two dates. The function subtracts TO_DATE from the FROM_DATE and places the difference into DAYS. All date fields are four-bytes packed.

```
DEF FROM_DATE     (P 7.0) = NONE 'FROM' 'DATE'
DEF TO_DATE       (P 7.0) = NONE 'TO' 'DATE'
DEF DAYS          (P 7.0) = NONE 'DAYS' 'BETWEEN'
DEF FUNCTION       (X 80) = NONE
```

```
SET FROM_DATE = CDATE
SET FUNCTION  = 'NUM_DAYS'

CALL CTSEDATE USING FUNCTION
                    FROM_DATE
                    TO_DATE
                    DAYS
```

SUB_DAYS Function—Subtract Days From a Date

Use the SUB_DAYS function to subtract a number of days from a date, giving a resultant date.

```
DEF FROM_DATE     (P 7.0) = NONE
DEF DAYS          (P 7.0) = NONE
DEF RESULT        (P 7.0) = NONE
DEF FUNCTION       (X 80) = NONE
SET FROM_DATE = EXPDT
SET DAYS      = 7
SET FUNCTION  = 'SUB_DAYS'

CALL CTSEDATE USING FUNCTION
                    FROM_DATE
                    DAYS
                    RESULT
```

SUB_YEARS Function—Subtract Years From a Date

Use the SUB_YEARS function to subtract a number of years from a date, giving a resultant date.

```
DEF FROM_DATE    (P 7.0) = NONE 'FROM' 'DATE'
DEF YEARS        (P 7.0) = NONE 'YEARS' 'SUBTRACTED'
DEF RESULT_DATE  (P 7.0) = NONE 'RESULT' 'DATE'
DEF FUNCTION      (X 80) = NONE
SET FROM_DATE = EXPDT
SET YEARS     = 3
SET FUNCTION  = 'SUB_YEARS'

CALL CTSEDATE USING FUNCTION
                    FROM_DATE
                    YEARS
                    RESULT_DATE
```

CA Earl Use

Sample JCL for running CA Earl can be found in member TMSEARL in the CA 1 *hlq*.CTAPJCL library. See the EARLINDX member in the CA 1 *hlq*.CTAPEARL library for a list of supplied sample members. The following coding sequence example illustrates the two requirements for using CA Earl:

1. Copy the EARLDEFS member using a COPY statement.
2. Set the type of run desired using the SET statement. (SEQUENTIAL is the default.)

In this example, CA Earl lists all data sets that expire after the current date. A SET statement has been used to define a date format different from the default that CA 1 system option DATEFMT defines. The CONTROL statement defines the sort sequence for the output report.

```
!  DEFINE WORK FIELDS
DEF PRINT_EXPDT      (X 10) = NONE 'EXPIRATION' 'DATE'
DEF PRINT_CURR_DATE  (X 10) = NONE 'TODAYS' 'DATE'
!  COPY CA-1 CA-EARL RECORD DEFINITIONS
COPY EARLDEFS
!  SET TYPRUN TO RETRIEVE VOLUME AND DSNB RECORDS
SET TYPRUN = R_SEQUENTIAL
!  ISSUE GET FOR TMC RECORD
!SET DATE FORMAT
SET DATE_FMT='MMMDD.YYYY'
GET TAPEDB
!  USE CTSEDATE TO FORMAT EXPDT AND RUNDATE FOR PRINTING
SET FUNCTION = 'TO_EXTERNAL'
CALL  CTSEDATE  USING FUNCTION
                PRINT_EXPDT
                EXPDT
                DATE_FMT
SET FUNCTION = 'RETURN_DATE'
CALL  CTSEDATE  USING FUNCTION
                PRINT_CURR_DATE
                RUNDATE
                DATE_FMT
!  QUIT WHEN WE REACH END-OF-FILE ON THE TMC
GOTO EOJ TAPEDB = 'E'
!  SELECT DATA SETS EXPIRING AFTER CURRENT DATE
REPORT 'DATA SETS EXPIRING BEYOND CURRENT DATE'
 SELECT EXPDT > RUNDATE
 CONTROL EXPDT DSN
 PRINT VOLSER DSN PRINT_CURR_DATE PRINT_EXPDT
 END
```

EARLDEFS Tape Record Field Definitions

The following CA Earl statements define the record passed to CA Earl. The TYPE notation indicates whether a field is character (X), binary (B), or packed decimal (P).

You must issue a COPY statement (COPY EARLDEFS) to bring these definitions into your CA Earl program from the CA 1 Source library.

Field Name	Position	Type	Title	Source
!				
! TAPE RECORD - COMMON AREA				
!				
DEF FORMAT_INDICATOR	001-001	B	'FORMAT' 'INDICATOR'	X 'FF'
DEF RECORD_TYPE	002-002	X	'RECORD' 'TYPE'	'1'
DEF PRODUCT_CODE	003-004	X	'PRODUCT' 'CODE'	'L0'
DEF PRODUCT_RELEASE	005-006	X	'PRODUCT' 'RELEASE'	'nn'
DEF VOLSER	007-012	X	'VOLSER'	TMVOLSER OR DSNBVSN
DEF VOLSEQ	013-014	B	'VOL' 'SEQ'	TMVOLSEQ
DEF BASEVOL	015-020	X	'BASE' 'VOLSER'	TMFRSVOL/TMVOLSER/DSNBVSN
DEF FILESEQ	021-022	B	'FILE' 'SEQ'	DSNBFSN
DEF FILECNT	023-024	B	'FILE' 'CNT'	TM#DSNBS + 1
DEF EXPDT	025-028	P	'EXPIRATION' 'DATE'	TMEXPDT
DEF DSN	029-072	X	'DATA SET NAME'	TMDSN OR DSNBDSN
DEF RECFM	073-076	X	'REC' 'FMT'	TMRECFM OR DSNBRFM(XLATED)
DEF LRECL	077-080	B	'REC' 'LEN'	TMLRECL OR DSNBLRECL
DEF BLKSIZE	081-084	B	'BLOCK' 'SIZE'	TMBLKSI DSNBBLKS
DEF BLKCNT	085-088	B	'BLOCK' 'COUNT'	TMBLKCNT DSNBBLKC
DEF CDATE	089-092	P	'CREATE' 'DATE'	TMCRTDT DSNBCRDT
DEF CTIME	093-096	P	'CREATE' 'TIME'	TMCRTTI DSNBCRTM
DEF CJOB	097-104	X	'CREATE' 'JOBNAME'	TMJOBNM DSNBCJN
DEF CSTEP	105-112	X	'CREATE' 'STEP'	TMSTPNAM DSNBCSN
DEF CDDNAME	113-120	X	'CREATE' 'DDNAME'	TMDDNAME
DEF CUNIT	121-124	X	'CREATE' 'UNIT'	TMCRUNI XLATED

Field Name	Position	Type	Title	Source
DEF CLOSE_IND	125-125	X	'CLS' 'IND'	TMFLAG1 'C', 'O' OR 'A' *SEE NOTE
DEF LDATE	126-129	P	'LAST' 'DATE'	TMLASUSD
DEF LTIME	130-133	P	'LAST' 'TIME'	TMLASUST
DEF LJOB	134-141	X	'LAST' 'JOB'	TMLASUSJ
DEF LUNIT	142-145	X	'LAST' 'UNIT'	TMUSUNI XLATED
DEF COPYCAT_IND	146-146	X	'COPYCAT' 'IND'	TMFLAG3 'Y' OR 'N'
DEF DEN	147-150	X	'DEN'	TMDEN XLATED
DEF LABEL_TYPE	151-153	X	'LBL' 'TYP'	TMLTYPE XLATED
DEF SCRATCH_IND	154-154	X	'SCR' 'IND'	TMFLAG1 'Y' OR 'N'
DEF TRTCH	155-158	X	'TRTCH'	TMTRTCH XLATED
DEF USECLN	159-160	B	'USE' 'CLN'	TMUSECLN
DEF CLNCNT	161-162	B	'CLN' 'CNT'	TMCLNCNT
DEF DATECLN	163-166	P	'DATE' 'CLEANED'	TMDATCLN
DEF BTHDATE	167-170	P	'BIRTH' 'DATE'	TMBTHDAT
DEF PURDATE	167-170	P	'PURCHASE' 'DATE'	TMBTHDAT
DEF USECNT	171-172	B	'USE' 'CNT'	TMUCOUNT
DEF COUNT	171-172	B	'USE' 'CNT'	TMUCOUNT
DEF OUTDATE	173-176	P	'DATE' 'MOVED'	TMOUTDAT
DEF MOVEDATE	173-176	P	'DATE' 'MOVED'	TMOUTDAT
DEF OUTCODE	177-180	X	'LOC' 'ID'	TMOUTAR
DEF OUTLOC	177-180	X	'LOC' 'ID'	TMOUTAR
DEF STACKING_INUSE	181-181	X	'STK' 'USE'	TMFLAG4 TMINUSE Y/N
DEF STACKING_FULL	182-182	X	'STK' 'FUL'	TMFLAG4 TMNOSTAK Y/N
DEF STACKING_VOLUME	183-183	X	'STK' 'VOL'	TMFLAG5 TMSTACK Y/N
DEF VOL_PERCENTAGE	184-184	B	TMVOLP	
DEF FILE_PERCENTAGE	185-185	B	'FIL' 'USE'	TMFILEP or DSNBPERC
DEF COMPRESSION	186-186	B	'COMP' 'RATE'	TMCMPRS or DSNBCPRS
DEF TEMP_READ_CLEAN	187-188	B	'TREAD' 'CLEAN'	TMTRERRC

Field Name	Position	Type	Title	Source
DEF TEMP_WRITE_CLEAN	189-190	B	'TWRITE' 'CLEAN'	TMTWERRC
DEF PERM_READ_CLEAN	191-192	B	'PREAD' 'CLEAN'	TMPRERRC
DEF PERM_WRITE_CLEAN	193-194	B	'PWRITE' 'CLEAN'	TMPWERRC
DEF TEMP_READ_INIT	195-196	B	'TREAD' 'INIT'	TMTRERRI
DEF TEMP_WRITE_INIT	197-198	B	'TWRITE' 'INIT'	TMTWERRI
DEF PERM_READ_INIT	199-200	B	'PREAD' 'INIT'	TMPRERRI
DEF PERM_WRITE_INIT	201-202	B	'PWRITE' 'INIT'	TMPWERRI
DEF VENDOR	203-210	X	'VENDOR'	TMVENDOR
DEF SERVICE_IND	211-214	X	'SRV' 'IND'	TMFLAG3 ' ' or 'BAD'
DEF CPUID	215-218	X	'CPU' 'ID'	TMVACPU
DEF EDMID	219-226	X	'EDM' 'ID'	TMEDMID
DEF OWNER	219-226	X	'VOLUME' 'OWNER'	TMEDMID
DEF ACCT	227-286	X	'USER' 'DATA'	TMUSER
DEF USER_DATA	227-286	X	'USER' 'DATA'	TMUSER
DEF VOLUME_COUNT	287-288	B	'VOLUME' 'COUNT'	Not valid for CA 1
DEF CREATE_PROGRAM	289-296	X	'CREATE' 'PROGRAM'	TMCPGM DSNBCPGM
DEF LAST_USE_PROGRAM	297-304	X	'LAST' 'PROGRAM'	TMLPGM
DEF ROBOT_TYPE	305-312	X	'ROBOT' 'TYPE'	TMROBTY
DEF ROBOT_NUMBER	313-315	N	'ROBOT' 'NUM'	TMROBID
DEF ACTUAL_VOLSER	316-321	X	'ACTUAL' 'VOLSER'	CL6
DEF ACTUAL_VOL_IND	322-322	X	'ACV' 'IND'	TMFLAG4 'Y' OR 'N'
DEF SMS_CLASS	323-330	X	'SMS' 'CLASS'	TMSMSMC DSNBSMSM
DEF COMMON_RSRV0	331-331	X		
DEF VOL_DEGAUS	332-332	X	'VOL' 'DEG'	TMFLAG4 'Y' OR 'N'
DEF FILE_IS_CAT	333-333	X	'OS' 'CAT'	TMFLAG4 'Y' OR 'N'
VLT_SPEC_REQ	334-334	X	'VLT' 'REQ'	TMFLAG4 'Y' OR 'N'
NON_RES_IND	335-335	X	'NON' 'RES'	TMFLAG4 'Y' OR 'N'

Field Name	Position	Type	Title	Source
DEF SLOT	336-342	X	'SLOT'	TMSLOT
DEF WORM_ID	343-354	X	'WORM' 'ID'	PIC H
DEF WORM_COUNT	355-356	B	'WORM' 'CNT'	PIC'ZZZZZ'
DEF WORM_IND	357-357	X	'WORM' 'IND.'	
DEF BYTE PERCENTAGE	358-358	B	'BYTE' 'PERC'	PIC 'ZZZ'
DEF COMMON_RSRV	359-360	X		
DEF BES_KEY	361-364	B	'BES'	'KEY'
!				
! TAPE RECORD - TLMS SPECIFIC AREA				
!				
DEF TLMS_RESERVED	365-430	X		
DEF VOLUME_OUTSRV	431-434	x	'OUTSRV' 'CODES'	
DEF VOLUME_LOST_CODE	431-431	X	'LOST' 'CODE'	
DEF VOLUME_DAMG_CODE	432-432	X	'DAMG' 'CODE'	
DEF VOLUME_DEST_CODE	433-433	X	'DEST' 'CODE'	
DEF VOLUME_SPAR_CODE	434-434	X		
DEF VOLUME_DEST_DATE	435-438	p	'DESTROY' 'DATE' PIC 'ZZ99/999'	
DEF ACTIVE_FILES	439-440	B	'ACTIVE' 'FILES'	
DEF TAPE_LENGTH	441-442	B	'TAPE' 'LENGTH'	
DEF TAPE_TYPE	443-444	X	'TAPE' 'TYPE'	
DEF USES_CERTIFY	445-446	B	'USES' 'CERTIFIED'	
DEF TIMES_CERTIFY	447-448	B	'TIMES' 'CERTIFIED'	
DEF DATE_CERTIFY	449-452	P	'DATE' 'CERTIFIED'	
DEF CDS_FILE_SEQ	453-454	B	'CDS FILE' 'SEQUENCE'	
DEF DATE_SCRATCHED	455-458	P	'DATE' 'SCRATCHED'	
DEF RTN_DATA	459-490	X	'RETENTION' 'DATA'	

Field Name	Position	Type	Title	Source
DEF DSN_KEEP_DATE	491-494	P	'DSN KEEP' 'DATE'	
DEF CREATE_PROGRAM	495-502	X	'CREATING' 'PROGRAM'	
DEF VOLUME_KEEP_DATE	503-506	P	'VOL KEEP' 'DATE'	
DEF VOLUME_EXPDT	507-510	P	'VOL' 'EXPDT'	
DEF SCRATCH_SOURCE	511-511	X	'SCRATCH' 'SOURCE'	
DEF SPARE_FLAG	512-512	X	'SPARE' 'FLAG'	
!				
! TAPE RECORD - CA-1 SPECIFIC AREA				
!				
DEF CA1_RSRV	365-380	X		
DEF FLAG1_SCR	381-381	X	'SCR' 'FLG'	TMFLAG1 'Y' or 'N'
DEF FLAG2_TEM	382-382	X	'TEM' 'IND'	TMFLAG2 'Y' or 'N'
DEF PREVVOL	383-388	X	'PREV' 'VOLUME'	TMPRVVOL
DEF NEXTVOL	389-394	X	'NEXT' 'VOLUME'	TMNXTVOL
DEF F1STVSN	395-400	X	'FIRST' 'VOLUME'	DSNBFVSN
DEF FIRST_DSNB	401-404	B	'FIRST' 'DSNB'	TMADSNB
DEF LSTDSNB	405-408	B	'LAST' 'DSNB'	TMALDSNB
DEF CURDSNB	409-412	B	'CURR' 'DSNB'	DSNBCURR
DEF PREV	413-416	B	'PREV' 'DSNB'	DSNBPREV
DEF NEXT	417-420	B	'NEXT' 'DSNB'	DSNBNEXT
DEF DSN17	421-437	X	'HDR1' 'DATA SET NAME'	TMDSN17
DEF UNUSED2	438-439	X		
DEF CATALOG_COUNT	440-440	B	'CAT' 'CNT'	TMCATCNT or DSNBCCNT
DEF FLAG1_RDS	441-441	X	'RDS' 'IND'	TMFLAG1 'Y' or 'N'
DEF FLAG1_DEL	442-442	X	'DEL' 'IND'	TMFLAG1 'Y' or 'N'
DEF FLAG1_CLN	443-443	X	'CLN' 'IND'	TMFLAG1 'Y' or 'N'
DEF FLAG1_UPD	444-444	X	'UPD' 'IND'	TMFLAG1 'Y' or 'N'
DEF FLAG1_INT	445-445	X	'INT' 'IND'	TMFLAG1 'Y' or 'N'
DEF FLAG2_ETM	446-446	X	'ETM' 'IND'	TMFLAG2 'Y' or 'N'

Field Name	Position	Type	Title	Source
DEF FLAG2_EDD	447-447	X	'EDD' 'IND'	TMFLAG2 'Y' or 'N'
DEF FLAG2_ECC	448-448	X	'ECC' 'IND'	TMFLAG2 'Y' or 'N'
DEF FLAG2_E99	449-449	X	'E99' 'IND'	TMFLAG2 'Y' or 'N'
DEF FLAG4_SMS	450-450	X	'SMS' 'IND'	TMFLAG4 'Y' OR 'N'
DEF FLAG1_TMS	451-451	X	'TMS' 'IND'	TMFLAG1 'Y' OR 'N'
DEF FLAG2_REC	452-452	X	'REC' 'IND'	TMFLAG2 'Y' or 'N'
DEF FLAG2_OUT	453-453	X	'OUT' 'IND'	TMFLAG2 'Y' or 'N'
DEF FLAG2_CAT	454-454	X	'CAT' 'IND'	TMFLAG2 'Y' or 'N'
DEF FLAG3_DYN	455-455	X	'DYN' 'IND'	TMFLAG3 'Y' or 'N'
DEF FLAG3_EDM	456-456	X	'EDM' 'IND'	TMFLAG3 'Y' or 'N'
DEF FLAG3_BAD	457-457	X	'BAD' 'IND'	TMFLAG3 'Y' or 'N'
DEF PROGRAM_ID	458-465	X	'PROGRAM' 'NAME'	BATCHID/HOOKID XLATED
DEF AUDATE	466-469	P	'AUDIT' 'DATE'	TMVADATE
DEF AUTIME	470-473	P	'AUDIT' 'TIME'	TMVATIME
DEF USERID	474-481	X	'USERID'	TMVAUSER
DEF AUCODE	482-482	X	'AUDIT' 'CODE'	TMVACODE 'Y' or 'N'
DEF CA1_RSRV2	483-485	X		
DEF ACTIND	486-486	X	'ACT' 'IND'	DSNBACT 'Y' or 'N' or 'S'
DEF EXCEPTION_CODES	487-489	X	'EXCEPTION' 'CODES'	
DEF VOLFLG1	490-490	B	'F1'	TMFLAG1
DEF VOLFLG2	491-491	B	'F2'	TMFLAG2
DEF VOLFLG3	492-492	B	'F3'	TMFLAG3
DEF VOLFLG4	493-493	B	'F4'	TMFLAG4
DEF VOLFLG5	494-494	B	'F5'	TMFLAG5
DEF VOLFLG6	495-495	B	'F6'	TMFLAG6
DEF CA1_RSRV3	496-496	X		
DEF DSNBFLG1	497-497	B	'F1'	DSNBFLG1
DEF DSNBFLG2	498-498	B	'F2'	DSNBFLG2
DEF CA1_RSRV4	499-500	X		
DEF EXTERNAL_VAULT	501-501	X	'EXTERNAL' 'VAULT'	TMFLAG3 'Y' OR 'N'

Field Name	Position	Type	Title	Source
DEF DATASET_ERASE	502-502	X	'DATASET' 'ERASE'	TMFLAG3 'Y' OR 'N'
DEF DEFAULT_EXPDT	503-503	X	'DEFAULT' 'EXPDT'	TMFLAG3 'Y' OR 'N'
DEF MULTI_FILES	504-504	X	'MULTI' 'FILES'	TMFLAG3 'Y' OR 'N'
DEF CA1_RSRV	505-512	X		

PREDEFINED DEFINITIONS

```
! TAPE DATABASE I/O (TYPRUN)
!   FUNCTIONS    DATA SETS                     IN ORDER BY
!   VOLUME     - FIRST ON VOL                  VOLSER
!   SEQUENTIAL - ALL                           VOLSER, FILESEQ
!   CHAINED    - ALL                           BASE VOLUME, FILESEQ
!   REPORT - REPORT FILE                       USAGE
!   VAULT - VAULT FILE                         USAGE
!   AUDIT - AUDIT FILE                         USAGE
!
    DEF R_VOLUME                                (X 10) = 'VOLUME    '
    DEF R_SEQUENTIAL                           (X 10) = 'SEQUENTIAL'
    DEF R_CHAINED                              (X 10) = 'CHAINED   '
    DEF R_REPORT                               (X 10) = 'REPORT    '
    DEF R_VAULT                                (X 10) = 'VAULT     '
    DEF R_AUDIT                                (X 10) = 'AUDIT     '
    DEF R_RETENTION                            (X 10) = 'RETENTION '
    DEF R_FILE                                 (X 10) = 'FILE      '
!
! DATE FUNCTIONS
! FUNCTION = RETURN_DATE
!       ADD_DAYS
!       ADD_YEARS
!       ADD_WORK
!       NUM_DAYS
!       TO_EXTERNAL
!       FROM_EXTERNAL
!       SET_FORMAT
!       SUB_DAYS
!       SUB_YEARS
!
   DEF DATE_FMT                      (X 10)  ='DEFAULT   '
   DEF FUNCTION                      (X 80)  = NONE
   DEF DAYS                          (P 7.0) = 0
   DEF YEARS                         (P 7.0) = 0
   DEF RUNDATE                       (P 7.0) = 0
   DEF RUNTIME                       (P 7.0) = 0
!
! DATE TYPES (TEST VALUES)
!
   DEF T_ZEROS                       (B 3.0) =  0
   DEF T_JDATE                       (B 3.0) =  4
   DEF T_OLD_JULIAN                  (B 3.0) =  8
   DEF T_FOREIGN                     (B 3.0) = 12
   DEF T_LDATE                       (B 3.0) = 16
   DEF T_CATLG                       (B 3.0) = 20
   DEF T_CYCLE                       (B 3.0) = 24
   DEF T_PERM                        (B 3.0) = 28
   DEF T_CATLG_DAYS                  (B 3.0) = 32
   DEF T_USER                        (B 3.0) = 36
   DEF T_STATS                       (B 3.0) = 40
   DEF T_MSG                         (B 3.0) = 44
   DEF T_AGE                         (B 3.0) = 48
!
! EDITED FORM OF DATE VARIABLES
!
```

DEF P_EXPDT	(X 10) = NONE 'EXPIRATION' 'DATE'
DEF P_CDATE	(X 10) = NONE 'CREATE' 'DATE'
DEF P_CTIME	(P 6.0) = NONE 'CREATE' 'TIME' PIC '99:99:99'
DEF P_LDATE	(X 10) = NONE 'LAST USE' 'DATE'
DEF P_LTIME	(P 6.0) = NONE 'LAST USE' 'TIME' PIC '99:99:99'
DEF P_DATECLN	(X 10) = NONE 'DATE' 'CLEANED'

PREDEFINED DEFINITIONS

DEF P_PURDATE	(X 10) = NONE 'PURCHASE' 'DATE'
DEF P_BTHDATE	= P_PURDATE 1-10 X 'PURCHASE' 'DATE'
DEF P_MOVEDATE	(X 10) = NONE 'DATE' 'MOVED'
DEF P_OUTDATE	= P_MOVEDATE 1-10 X 'DATE' 'MOVED'
DEF P_RUNDATE	(X 10) = NONE 'RUN' 'DATE'
DEF P_RUNTIME	(P 6.0) = NONE 'RUN' 'TIME' PIC '99:99:99'
DEF P_DATE_CERTIFY	(X 10) = NONE 'DATE' 'CERTIFIED'
DEF P_DATE_SCRATCHED	(X 10) = NONE 'DATE' 'SCRATCHED'
DEF P_DSN_KEEP_DATE	(X 10) = NONE 'DSN KEEP' 'DATE'
DEF P_VOLUME_KEEP_DATE	(X 10) = NONE 'VOL KEEP' 'DATE'
DEF P_VOLUME_EXPDT	(X 10) = NONE 'VOL EXPDT'
DEF P_AUDATE	(X 10) = NONE 'AUDIT' 'DATE'
DEF P_AUTIME	(P 6.0) = NONE 'AUDIT' 'TIME' PIC '99:99:99'
!	
! DATA BLOCK FOR TO_EXPLODED DATE FUNCTION	
!	
DEF P_EXP_DATA	(X 64) = NONE
DEF P_EXP_TYPDAT	= P_EXP_DATA 01-04 B 'TYPE OF' 'DATE'
DEF P_EXP_TYPNAM	= P_EXP_DATA 05-14 X 'TYPE OF' 'DATE'
DEF P_EXP_TYPALT	= P_EXP_DATA 15-20 X 'ALT' 'KEYWORD'
DEF P_EXP_DYNAM	= P_EXP_DATA 21-30 B 'DAY OF' 'WEEK'
DEF P_EXP_DAYABV	= P_EXP_DATA 31-33 X 'DAY OF' 'WEEK'
DEF P_EXP_DOW	= P_EXP_DATA 34-34 X 'DAY OF' 'WEEK'

DEF P_EXP_DAYTYP	= P_EXP_DATA 35-35 X 'DAY' 'TYPE'
DEF P_EXP_LYR	= P_EXP_DATA 36-36 X 'LEAP' 'YEAR'
DEF P_EXP_PAKDDD	= P_EXP_DATA 37-40 P 'JULIAN' 'DATE'
DEF P_EXP_MONNAM	= P_EXP_DATA 41-50 X 'MONTH'
DEF P_EXP_MONABV	= P_EXP_DATA 51-53 X 'MONTH'
DEF P_EXP_ALLDAT	= P_EXP_DATA 54-64 X 'ALL' 'DATE'
DEF P_EXP_MM	= P_EXP_ALLDAT 01-02 X 'MONTH'
DEF P_EXP_DD	= P_EXP_ALLDAT 03-04 X 'DAY'
DEF P_EXP_YYYY	= P_EXP_ALLDAT 05-08 X 'YEAR'
DEF P_EXP_DDD	= P_EXP_ALLDAT 09-11 X 'JULIAN' 'DAY'

Note: C = CLOSE, O = OPEN, A = ABEND

TMEYDEFR Tape Record Field Definitions

The following CA Earl. statements define the record passed to CA Earl.The TYPE notation indicates whether a field is character (X), binary (B), or packed decimal (P).

FIELD NAME	POSITION	TYPE	TITLE	SOURCE
!				
! TAPE RECORD - CA-1 SPECIFIC AREA FOR				
! TYPERUN = 'RETENTION'				
!				
DEF RTN_OLD_EXPDT	513-516	P	'OLD' 'EXPDT'	TMSCLEAN, TMSCYCLE
DEF RTN_CLEAN_DATE	517-520	P	'ORIGINAL' 'CLEAN DATE'	TMSCLEAN
DEF RTN_CLNIN	521-525	X	'CLEAN' 'INDICATOR'	TMSCLEAN
DEF RTN_UNCIND	526-530	X	'UNCATALOG' 'INDICATOR'	TMSCLEAN, TMSCTLG
DEF RTN_SCRPOOL	531-543	X	'SCRATCH' 'POOL'	TMSCLEAN, TMSEXPDT

TMEYDEFV Tape Record Field Definitions

The following CA Earl statements define the record passed to CA Earl. The TYPE notation indicates whether a field is character (X), binary (B), or packed decimal (P).

FIELD NAME	POSITION	TYPE	TITLE	SOURCE
!				
! TAPE RECORD - CA-1 SPECIFIC AREA FOR				
! TYPERUN = 'VAULT'				
!				
DEF VAULT_FROM_AREA	513-516	X	'FROM' 'AREA'	TMSVMVLT
DEF VAULT_FROM_SLOT	517-520	B	'FROM' 'SLOT'	TMSVMVLT
DEF VAULT_PATT_SEQ	521-524	B	'VAULT' 'PATTERN'	TMSVMVLT
DEF VAULT_DSN_SEQ	525-528	B	'DSN' 'SEQUENCE'	TMSVMVLT
DEF VAULT_SIZE	529-532	B	'VAULT' 'SIZE'	TMSVMVLT
DEF VAULT_FROM_SUBCODE	533-533	X	'FROM' 'SUB'	TMSVMVLT
DEF VAULT_OUT_SUBCODE	534-534	X	'OUT' 'SUB'	TMSVMVLT
DEF VAULT_LBLTYP	535-535	X	'LABEL' 'TYPE'	TMSVMVLT
DEF VAULT_GDG_DISPL	536-536	B	'GDG' 'DISP'	TMSVMVLT
DEF VAULT_CJOB	537-544	X	'CREATION' 'JOB'	TMSVMVLT
DEF VAULT_CDATE	545-548	P	'CREATION' 'DATE'	TMSVMVLT
DEF VAULT_CTIME	549-551	P	'CREATION' 'TIME'	TMSVMVLT
DEF VAULT_LCODE	552-552	X	'PSEUDO' 'CODE'	TMSVMVLT
DEF VAULT_GEN_VERS	553-561	X	'GENERATION' 'VERSION'	TMSVMVLT

CA Earl Coding Examples

Example 1

This example illustrates how to use CA Earl to compute the optimum tape block size for each data set, and produce a report that lists the data sets whose block size is not within 25 percent of the optimum block size. A similar example resides as member TMEBLKS in the CA 1 Source library.

```
!
!  TMS REPORT-TMEBLKS   DSNS NOT WITHIN 25% OF OPTIMUM BLKSIZE (32K)
!     ( SIMILAR TO OLD EARL01 REPORT )
!        VOLUME AND DSNB RECORDS
!

!
!  USE CA-Earl TO GENERATE A REPORT OF DSN'S WHOSE BLKSIZES
!           ARE NOT WITHIN 25% OF OPTIMUM (32760).
!

 DEF WORK1      (B 07.3) = 0.000
 DEF P_OPT      (B 09.1) = 0.0   '% OF' 'OPTIMUM'
 DEF WORK3      (B 10.0) = 0     'OPTIMUM' 'BLKSIZE'
 DEF FIRST_SW      (X 1) = ' '

 COPY EARLDEFS

 SET TYPRUN = R_SEQUENTIAL

 GET TAPEDB
 GOTO EOJ TAPEDB = 'E'

 IF SCRATCH_IND = 'N' OR ACTIND = 'Y' THEN
     GOTO CONTINUE
   ELSE
     GOTO START
 ENDIF

 CONTINUE:
 IF FIRST_SW = ' '
    THEN SET FIRST_SW = '*'
    SET FUNCTION = 'RETURN_DATE'
    CALL CTSEDATE USING FUNCTION,RUNDATE,RUNTIME
    SET FUNCTION = 'TO_EXPLODED'
    CALL CTSEDATE USING FUNCTION,P_EXP_DATA,RUNDATE
 ENDIF
```

(continued on next page)

```
!
!  COMPUTE OPTIMUM BLKSIZE (BASED ON 32760 AND LRECL)
!

 IF LRECL > 0 THEN
    SET WORK3 = 32760 / LRECL
    SET WORK3 = WORK3 * LRECL
 ELSE
    GOTO START
 ENDIF

!
!  COMPUTE PERCENTAGE OF OPTIMUM
!

 IF BLKSIZE > 0 THEN
    SET WORK1 = BLKSIZE
    SET WORK1 = (WORK1 / WORK3)
    SET WORK1 = WORK1 * 100
    SET P_OPT = WORK1
 ELSE
    GOTO START
 ENDIF

 SET   FUNCTION  = 'TO_EXTERNAL'
 CALL  CTSEDATE  USING FUNCTION,P_CDATE,CDATE,DATE_FMT


 REPORT TITLE  @1 P_EXP_DAYNAM @11 ',' @13 P_EXP_MONNAM @24 P_EXP_DD
               @26 ',' @28 P_EXP_YYYY @32 '.' @33 P_EXP_DDD
               @50 COMPANY_NAME @115 'PAGE' @125 PAGENUMBER
        TITLE  @1 ' '
        TITLE  @1 'TMS REPORT-TMEBLKS'
                3 'DSNS NOT WITHIN 25% OF OPTIMUM BLKSIZE (32K)'

 SELECT  WORK1 < 75
 CONTROL DSN VOLSEQ
 PRINT   VOLSER VOLSEQ FILESEQ DSN BLKSIZE WORK3 P_OPT LRECL CJOB
         P_CDATE

END
```

```
 THURSDAY, JULY 21, 2005.200                  Y O U R   C O M P A N Y   N A M E                  PAGE      1

TMS REPORT-TMEBLKS   DSNS NOT WITHIN 25% OF OPTIMUM BLKSIZE (32K)
-----------------------------------------------------------------------------------------------------------------------------------
VOLSER   VOL  FILE DATA SET NAME                                 BLOCK     OPTIMUM        % OF          REC CREATE   CREATE
         SEQ   SEQ                                                SIZE     BLKSIZE      OPTIMUM         LEN JOBNAME  DATE
-------------------------------------------------------------------------------------------------------------------------------------
374477    1     1  AZ.M1000MX1                                   12144      32736        37.0          44  RESRCJMA YYYY/224
370548    1     1  AZ.M1000MX2                                   12000      32720        36.6          80  RESRCJMA YYYY/224
328510    1     1  AZ.M1111ML1                                   12375      32670        37.8         165  RESRCJMA YYYY/260
366971    1     1  AZ.M1323ML1                                   12000      32720        36.6          80  RESRCJMA YYYY/142
328074    1     1  AZ.M1343ML1                                   12000      32720        36.6          80  RESRCJMA YYYY/260
370226    1     1  AZ.M1452ML0                                   12122      32538        37.2         319  RESRCJAB YYYY/095
344443    1     1  AZ.M1552ML9                                   12122      32538        37.2         319  RESRCJMA YYYY/260
336716    1     1  AZ.OPS60ML1                                    6160      32720        18.8          80  OPSCCT60 YYYY/013
332857    1     1  AZ.OPS66ML1                                   16250      32500        50.0         325  OPSCCJ66 YYYY/056
384019    1     1  AZ.OPS66ML1                                   16250      32500        50.0         325  OPSCCJ60 YYYY/015
326347    1     1  AZ.OPS66ML1                                   16250      32500        50.0         325  OPSCCT60 YYYY/013
360133    1     1  AZ.OPS69ME1                                    1596      32718         4.8         133  OPSCCJ66 YYYY/056
330975    1     1  AZ.USR39ML9                                   23500      31725        74.0        1175  USRCCJ40 YYYY/082
355137    1     1  AZ.USR39ML9                                   23500      31725        74.0        1175  USRCCJ40 YYYY/021
332588    1     1  AZ.YES31MLA                                   23500      31725        74.0        1175  YESCCJ42 YYYY/237
371005    1     1  AZ.YES31MLA                                    4900      32340        15.1         980  YESCCJTY YYYY/322
321313    1     1  AZ.YES31MLA                                   23500      31725        74.0        1175  YESCCJ31 YYYY/241
322214    1     1  AZ.YES31MLA                                   23500      31725        74.0        1175  YESCCJ31 YYYY/259
326002    1     1  AZ.YES31MLA                                   23500      31725        74.0        1175  YESCCJ31 YYYY/259
321112    1     1  AZ.YES42MLA                                   23500      31725        74.0        1175  YESCCJ42 YYYY/151
322212    1     1  AZ.YES42MLA                                   23500      31725        74.0        1175  YESCCJ55 YYYY/087
338672    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ50 YYYY/012
338200    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ50 YYYY/012
331088    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ50 YYYY/012
331512    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ80 YYYY/191
339720    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ50 YYYY/083
332275    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ50 YYYY/079
335502    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ50 YYYY/083
336981    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ50 YYYY/008
334254    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ55 YYYY/200
334686    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ50 YYYY/079
325864    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ50 YYYY/343
323562    1     1  AZ.YES50MLA                                   23500      31725        74.0        1175  YESCCJ50 YYYY/010
332931    1     1  AZ.YES50ML9                                    5875      31725        18.5        1175  YESCCJ99 YYYY/150
335467    1     1  AZ.YES50ML9                                    5875      31725        18.5        1175  YESCCJ99 YYYY/010
337262    1     1  AZ.YES50ML9                                    5875      31725        18.5        1175  YESCCJ99 YYYY/135
380155    1     1  AZ.YES55ME1                                    1596      32718         4.8         133  YESCCJ55 YYYY/090
331293    1     1  AZ.YES55ME1                                    1596      32718         4.8         133  YESCCJ55 YYYY/087
329444    1     1  BR.D11AHM8E                                   15464      30928        50.0        3866  GSEC0J6A YYYY/009
330814    1     1  BR.D12AHM8G                                   15464      30928        50.0        3866  GSEC0J6A YYYY/022
332400    1     1  BR.D12AHM9E                                   15848      31696        50.0        7924  GSEC0J6A YYYY/021
335247    1     1  BR.D12AHM9F                                   15848      31696        50.0        7924  GSEC0J6A YYYY/018
323274    1     1  BR.D12AHM9G                                   15848      31696        50.0        7924  GSEC0TFX YYYY/253
322289    1     1  BR.D12AHN02                                   15464      30928        50.0        3866  PCTLBJ6X YYYY/291
325568    1     1  BR.D12AHN07                                   15848      31696        50.0        7924  PCTX0J67 YYYY/276
328000    1     1  BR.D12BACK1.G0038V00                          21725      31600        68.7        1975  CORP0JMV YYYY/252
322201    1     1  BR.D12BACK1.G0039V00                          21725      31600        68.7        1975  CORP0JTC YYYY/252
334852    1     1  BR.D12BACK1.G0040V00                          21725      31600        68.7        1975  CORP0JH2 YYYY/255
330368    1     1  BR.D12BACK1.G0041V00                          21725      31600        68.7        1975  CORP0J31 YYYY/281
328831    1     1  BR.D22ATM95                                    9432      32226        29.2         786  CORP0MEN YYYY/040
```

Example 1 - Data Sets Not Within 25 Percent of Optimum Block

Example 2

This example illustrates how CA Earl can generate a volume serial master list. A similar example resides as member TMEVSNM in the CA 1 Source library.

```
!
!  TMS REPORT-TMEVSNM   VOLUME SERIAL MASTER
!     ( SIMILAR TO OLD TMS REPORT 1 )
!       VOLUME AND/OR RECORDS ONLY
!
!
!-----------------------  MAINTENANCE LOG  -------------------------
!
! ADD TOTALS TO REPORTS AND GENERAL ENHANCEMENTS             TE9A022
!
 COPY EARLDEFS

 DEF D_LBLTYPE         (X 4) = NONE 'LBL' 'TYPE'
 DEF D_VOLSER          (X 6) = NONE 'VOLUME' 'SERIAL'
 DEF D_CTIME           (P 4.0) = NONE 'CREATE' 'TIME' PIC '99:99'
 DEF P_FLAG1           (B 3.0) = 0    'F1' PIC H
 DEF FIRST_SW          (X 1) = ' '
 DEF VMS_INFO          (X 13) = ' '
   DEF VMS_CODE          =VMS_INFO 001-004 X
   DEF VMS_DATE          =VMS_INFO 006-013 X
 DEF NUM_DATASETS      (X 13) = ' '
   DEF NUM_TEXT        = NUM_DATASETS 001-009 X
   DEF NUM_FILES       = NUM_DATASETS 011-013 N PIC '999'
 DEF P_COMMENT         (X 13) = ' ' 'COMMENTS'
 DEF VOL_COUNT         (N 6.0) = 0 'VOLUMES'
 DEF DSNB_COUNT        (N 6.0) = 0 'DSNBS'

!
! USING A TYPRUN OF R_SEQUENTIAL WILL SELECT VOLUME AND DSNB RECORDS
! USING A TYPRUN OF R_VOLUME WILL SELECT ONLY VOLUME RECORDS
! USING A TYPRUN OF R_CHAINED WILL SELECT BASE VOLUME, THEN DSNB'S
!

!SET TYPRUN = R_SEQUENTIAL
!SET TYPRUN = R_VOLUME
 SET TYPRUN = R_CHAINED

 GOTO EOJ TAPEDB = 'E'
 GET TAPEDB
 GOTO TEST TAPEDB = 'E'

 IF FILESEQ = 0                   ! Q. UNALLOCATED DSNB RECORD
     GOTO START                   ! A. EQUAL - YES, BYPASS RECORD
 ENDIF

 IF ACTIND = 'N' AND FILESEQ > 1  ! Q. UNALLOCATED DSNB RECORD
     GOTO START                   ! A. EQUAL - YES, BYPASS RECORD
 ENDIF

 IF FLAG1_DEL = 'Y'               ! Q. TAPE IN DELETED STATUS
     GOTO START                   ! A. EQUAL - YES, BYPASS RECORD
 ENDIF
```

(continued on next page)

```
!IF CJOB = 'PRE-TMS'              ! Q. TAPE IN "PRE" STATUS
!    GOTO START                   ! A. EQUAL - YES, BYPASS RECORD
!ENDIF

!IF SCRATCH_IND = 'Y'             ! Q. TAPE IN SCRATCH STATUS
!    GOTO START                   ! A. EQUAL - YES, BYPASS RECORD
!ENDIF

 CONTINUE:
 IF FIRST_SW = ' '
    THEN SET FIRST_SW = '*'
    SET FUNCTION = 'RETURN_DATE'
    CALL CTSEDATE USING FUNCTION,RUNDATE,RUNTIME
    SET FUNCTION = 'TO_EXPLODED'
    CALL CTSEDATE USING FUNCTION,P_EXP_DATA,RUNDATE
 ENDIF

 SET FUNCTION = 'TO_EXTERNAL'
 CALL  CTSEDATE  USING FUNCTION,P_EXPDT,EXPDT,DATE_FMT
 CALL  CTSEDATE  USING FUNCTION,P_CDATE,CDATE DATE_FMT
 CALL  CTSEDATE  USING FUNCTION,P_LDATE,LDATE,DATE_FMT

 SET D_CTIME  = CTIME / 100
 SET D_LBLTYPE = LABEL_TYPE

 IF FILESEQ > 1 THEN
     SET P_FLAG1  = DSNBFLG1
     SET D_VOLSER = '      '
     SET D_LBLTYPE = 'DSNB'
     SET DSNB_COUNT = DSNB_COUNT + 1
   ELSE
     SET P_FLAG1  = VOLFLG1
     SET D_VOLSER = VOLSER
     SET VOL_COUNT = VOL_COUNT + 1
 ENDIF

 IF OUTCODE = '    ' OR OUTCODE = ' VMS' OR OUTCODE < '    '
     SET P_COMMENT = '             '
   ELSE
     SET VMS_CODE = OUTCODE
     CALL CTSEDATE USING FUNCTION,VMS_DATE,OUTDATE,DATE_FMT
     SET P_COMMENT = VMS_INFO
 ENDIF

 IF FILECNT > 1 AND FILESEQ = 1 THEN
     SET NUM_TEXT  = 'DATASETS='
     SET NUM_FILES = FILECNT
     SET P_COMMENT = NUM_DATASETS
 ENDIF

 IF SCRATCH_IND = 'Y' THEN
     SET P_COMMENT = '*SCRATCH*    '
 ENDIF

 IF FLAG3_BAD = 'Y' THEN
     SET P_COMMENT = '*BAD*        '
 ENDIF
```

(continued on next page)

```
REPORT TITLE  @1 P_EXP_DAYNAM @11 ',' @13 P_EXP_MONNAM @24 P_EXP_DD
              @26 ',' @28 P_EXP_YYYY @32 '.' @33 P_EXP_DDD
              @50 COMPANY_NAME @115 'PAGE' @125 PAGENUMBER
       TITLE  @1 ' '
       TITLE  @1 'TMS REPORT-TMEVSNM VOLUME SERIAL MASTER'

SELECT TAPEDB NOT = 'E'

CONTROL VOLSER VOLSEQ FILESEQ

 PRINT D_VOLSER 1 DSN 1 CJOB VOLSEQ FILESEQ 1 P_CDATE
     D_CTIME 1 P_EXPDT 1 P_LDATE D_LBLTYPE P_COMMENT
     P_FLAG1

 REPORT TITLE  @1 P_EXP_DAYNAM @11 ',' @13 P_EXP_MONNAM @24 P_EXP_DD
               @26 ',' @28 P_EXP_YYYY @32 '.' @33 P_EXP_DDD
               @50 COMPANY_NAME @115 'PAGE' @125 PAGENUMBER
        TITLE  @1 ' '
        TITLE  @1 'TMS REPORT-TMEVSNM  TOTALS FOR VOLUME SERIAL MASTER'
 SELECT TAPEDB = 'E'
 PRINT 0 'TOTAL NUMBER OF ITEMS LISTED:' VOL_COUNT 5 DSNB_COUNT
END
```

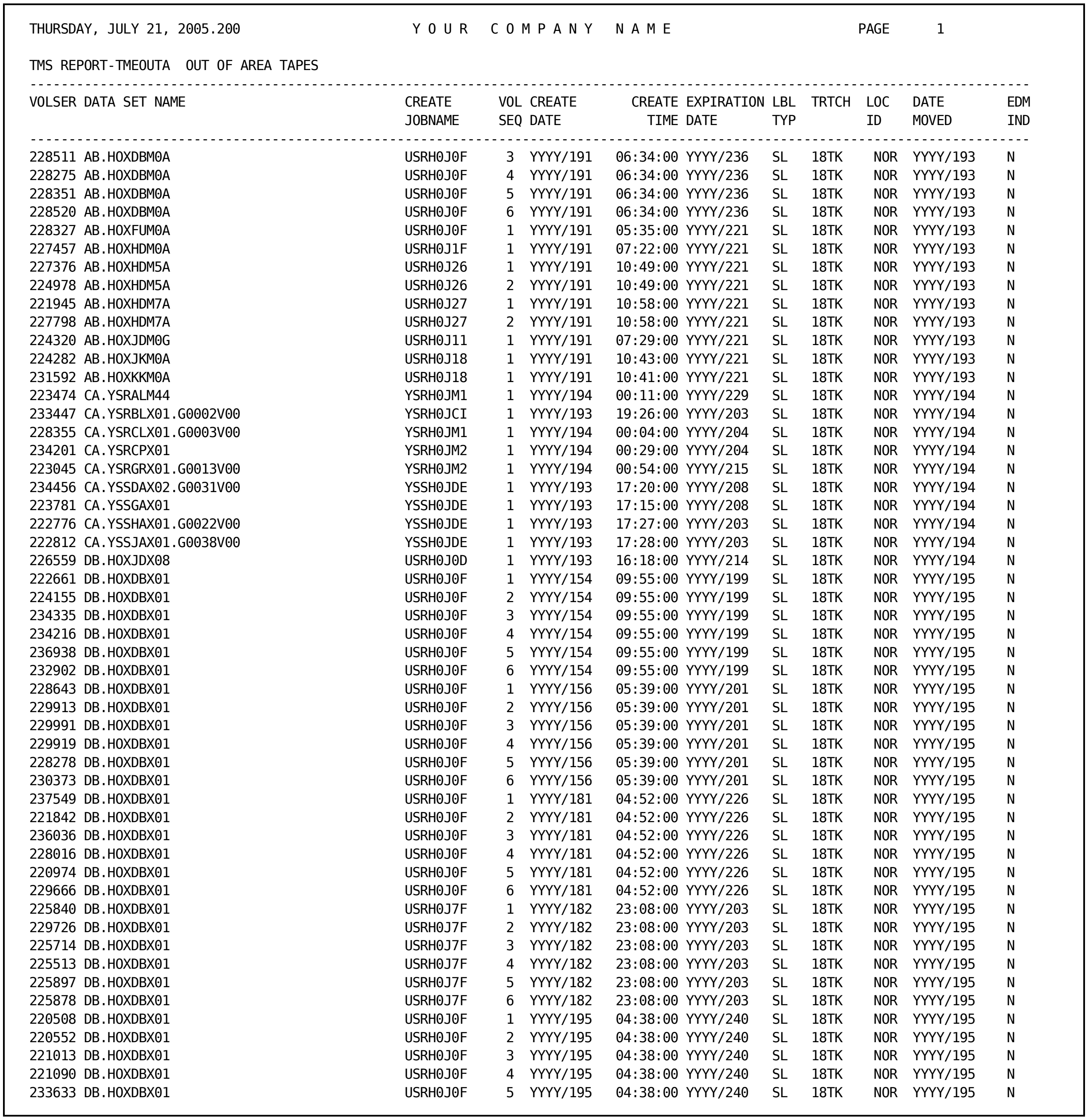
```
WEDNESDAY, JUNE 01, 2005.157                      Y O U R   C O M P A N Y   N A M E                        PAGE     49

 TMS REPORT-TMEVSNM  VOLUME SERIAL MASTER
 --------------------------------------------------------------------------------------------------------------------------------
 VOLUME DATA SET NAME                            CREATE    VOL FILE CREATE     CREATE EXPIRATION LAST USE   LBL  COMMENTS      F1
 SERIAL                                          JOBNAME   SEQ  SEQ DATE         TIME DATE       DATE       TYPE
 --------------------------------------------------------------------------------------------------------------------------------

 102724 ________________________________________ PRE-TMS     1    1 ZEROS       00:00 MMMDD YYYY ZEROS      SL   *SCRATCH*     44
 102725 ________________________________________ PRE-TMS     1    1 ZEROS       00:00 MMMDD YYYY ZEROS      SL   *SCRATCH*     44
 102726 ________________________________________ PRE-TMS     1    1 ZEROS       00:00 MMMDD YYYY ZEROS      SL   *SCRATCH*     44
 102728 ________________________________________ PRE-TMS     1    1 ZEROS       00:00 MMMDD YYYY ZEROS      SL   *SCRATCH*     44
 120000 ________________________________________ PRE-TMS     1    1 ZEROS       00:00 MMMDD YYYY ZEROS      SL   *SCRATCH*     44
 980000 GDS.BLC.BLC958.G0012V00                  SAMEJOB     1    1 MMMDD YYYY  04:27 MMMDD YYYY MMMDD YYYY SL   LIB2 MMMDD 19 A0
 980003 GDS.BLC.BLC959.G0013V00                  SAMEJOB     1    1 MMMDD YYYY  21:19 MMMDD YYYY MMMDD YYYY SL   MRON MMMDD 19 20
 980004 GDS.BLC.BLC959.G0013V00                  SAMEJOB     2    1 MMMDD YYYY  21:19 CATALOG    MMMDD YYYY SL   MRON MMMDD 19 A0
 980005 GDS.BLC.BLC959.G0013V00                  SAMEJOB     3    1 MMMDD YYYY  21:19 CATALOG    MMMDD YYYY SL   MRON MMMDD 19 A0
 980006 GDS.BLC.BLC578.G0012V00                  SAMEJOB     1    1 MMMDD YYYY  03:57 MMMDD YYYY MMMDD YYYY SL                 20
 980007 GDS.BLC.BLC578.G0012V00                  SAMEJOB     2    1 MMMDD YYYY  03:57 CATALOG    MMMDD YYYY SL                 20
 980008 GDS.BLC.BLC578.G0012V00                  SAMEJOB     3    1 MMMDD YYYY  03:57 CATALOG    MMMDD YYYY SL                 20
 980009 GDS.BLC.BLC578.G0013V00                  SAMEJOB     1    1 MMMDD YYYY  20:50 MMMDD YYYY ZEROS      SL   *SCRATCH*     24
 980010 NOBGE01.TMS52.DDRTESTI                   GLNGENER    1    1 MMMDD YYYY  12:31 MMMDD YYYY MMMDD YYYY SL                 E0
 980011 GDS.BLC.BLC958.G0014V00                  ANOTHER     1    1 MMMDD YYYY  14:09 MMMDD YYYY MMMDD YYYY SL   *SCRATCH*     24
 980012 HICFR01.AIVS.V102021.V980012.FILE1XXX    JOB0128A    1    1 MMMDD YYYY  11:09 MMMDD YYYY MMMDD YYYY SL                 61
 980013 ________________________________________ PRE-TMS     1    1 ZEROS       00:00 MMMDD YYYY ZEROS      SL   *SCRATCH*     44
 980014 HICFR01.AIVSGDG.G0002V00.DONTUSE         JOB0128H    1    1 MMMDD YYYY  13:16 MMMDD YYYY MMMDD YYYY SL   DATASETS= 009 21
        TEST.AGGR.FILE2                          JOBAGGR     1    2 MMMDD YYYY  19:12 MMMDD YYYY MMMDD YYYY DSNB               20
        TEST.AGGR.FILE3                          JOBAGGR     1    3 MMMDD YYYY  19:12 MMMDD YYYY MMMDD YYYY DSNB               20
        TEST.AGGR.FILE4                          JOBAGGR     1    4 MMMDD YYYY  19:12 MMMDD YYYY MMMDD YYYY DSNB               20
        TEST.AGGR.FILE5                          JOBAGGR     1    5 MMMDD YYYY  19:12 MMMDD YYYY MMMDD YYYY DSNB               20
        TEST.AGGR.FILE6                          JOBAGGR     1    6 MMMDD YYYY  19:12 MMMDD YYYY MMMDD YYYY DSNB               20
        TEST.AGGR.FILE7                          JOBAGGR     1    7 MMMDD YYYY  19:12 MMMDD YYYY MMMDD YYYY DSNB               20
        TEST.AGGR.FILE8                          JOBAGGR     1    8 MMMDD YYYY  19:12 MMMDD YYYY MMMDD YYYY DSNB               20
        TEST.AGGR.FILE9                          JOBAGGR     1    9 MMMDD YYYY  19:12 MMMDD YYYY MMMDD YYYY DSNB               20
 980015 HICFR01.AIVSGDG.G0013V00                 JOB0128M    1    1 MMMDD YYYY  14:59 CATALOG    MMMDD YYYY SL                 01
 980016 HICFR01.AIVSGDG.G0014V00.DONTUSE         JOBSLTAP    1    1 MMMDD YYYY  15:11 MMMDD YYYY MMMDD YYYY SL   *SCRATCH*     65
 980017 NOBGE01.TMS52.EOVDSA1                    GLNEOVA1    1    1 MMMDD YYYY  16:42 MMMDD YYYY MMMDD YYYY SL                 E0
 980018 NOBGE01.TMS52.EOVDSA1                    GLNEOVA1    2    1 MMMDD YYYY  16:42 MMMDD YYYY MMMDD YYYY SL                 A0
 980019 NOBGE01.TMS52.EOVDSA1                    GLNEOVA1    3    1 MMMDD YYYY  16:42 MMMDD YYYY MMMDD YYYY SL                 A0
 980020 NOBGE01.TMS52.TAPE01                     GLNCAT01    1    1 MMMDD YYYY  12:32 MMMDD YYYY MMMDD YYYY SL                 E0
 980021 NOBGE01.TMS52.TAPE02                     GLNCAT01    1    1 MMMDD YYYY  12:33 MMMDD YYYY MMMDD YYYY SL                 E0
 980022 NOBGE01.TMS52.TAPE03                     GLNCAT01    1    1 MMMDD YYYY  12:33 MMMDD YYYY MMMDD YYYY SL                 E0
 980023 SYS4.TEST.FOREIGN1                       MYJOB       1    1 MMMDD YYYY  10:46 MMMDD YYYY MMMDD YYYY SL                 E0
 980024 HICFR01.NOTAVIS.JOB01300.FILE1           JOB01300    1    1 MMMDD 1996  20:14 CATALOG    MMMDD YYYY SL                 E1
 980025 GDS.BLC.BLC958.G0014V00                  SAMEJOB     1    1 MMMDD YYYY  21:10 MMMDD YYYY ZEROS      SL   *SCRATCH*     24
 980026 HICFR01.NOTAVIS.JOB01300.FILE1           JOB01300    2    1 MMMDD YYYY  20:14 CATALOG    MMMDD YYYY SL                 41
 980027 HICFR01.NOTAVIS.JOB01300.FILE1           JOB01300    3    1 MMMDD YYYY  20:14 CATALOG    MMMDD YYYY SL   DATASETS= 002 41
        HICFR01.NOTAVIS.JOB0130Q.FILE2           JOB0130Q    3    2 MMMDD YYYY  20:28 CATALOG    MMMDD YYYY DSNB               8B
 980028 HICFR01.NOTAVIS.JOB01300.FILE1           JOB01300    4    1 MMMDD YYYY  20:14 CATALOG    MMMDD YYYY SL   DATASETS= 002 41
 980031 ________________________________________ JOBA131M    1    1 MMMDD YYYY  12:12 MMMDD YYYY MMMDD YYYY NL   *SCRATCH*     E5
 980032 SYS4.TEST.SHOW                           PAKING      1    1 MMMDD YYYY  15:20 MMMDD YYYY MMMDD YYYY SL                 E0
 980034 VAULT.TEST.AX.MF                         VAULTEST    1    1 MMMDD YYYY  03:01 MMMDD YYYY MMMDD YYYY SL   *SCRATCH*     A4
 980035 VAULT.TEST.AX                            VAULTEST    1    1 MMMDD YYYY  03:01 MMMDD YYYY MMMDD YYYY SL   *SCRATCH*     A4
 980036 VAULT.TEST.AX                            VAULTEST    1    1 MMMDD YYYY  03:01 MMMDD YYYY MMMDD YYYY SL   *SCRATCH*     A4
 980037 VAULT.TEST.AX                            VAULTEST    1    1 MMMDD YYYY  17:43 MMMDD YYYY MMMDD YYYY NL   *SCRATCH*     E5
 980038 VAULT.TEST.AX                            VAULTEST    1    1 MMMDD YYYY  11:54 MMMDD YYYY MMMDD YYYY SL   *SCRATCH*     E5
 980039 VAULT.TEST.AX                            VAULTEST    1    1 MMMDD YYYY  00:00 MMMDD YYYY MMMDD YYYY SL   *SCRATCH*     A4

 END OF REPORT
```

Example 2 - CA-Earl Volume Serial Number Master

Example 3

This example illustrates how CA Earl can generate a report of tapes that are out-of-area. A similar example resides as member TMEOUTA in the CA 1 Source library.

```
!
!  TMS REPORT-TMEOUTA   OUT-OF-AREA TAPES
!     ( SIMILAR TO OLD TMS REPORT 7 )
!         VOLUME RECORDS ONLY
!
 COPY EARLDEFS

 DEF FIRST_SW          (X 1) = ' '

 SET TYPRUN = R_VOLUME

 GET TAPEDB
 GOTO EOJ TAPEDB = 'E'

 IF FIRST_SW = ' '
    THEN SET FIRST_SW = '*'
    SET FUNCTION = 'RETURN_DATE'
    CALL CTSEDATE USING FUNCTION,RUNDATE,RUNTIME
    SET FUNCTION = 'TO_EXPLODED'
    CALL CTSEDATE USING FUNCTION,P_EXP_DATA,RUNDATE
 ENDIF

 SET FUNCTION = 'TO_EXTERNAL'
 CALL  CTSEDATE  USING FUNCTION,P_EXPDT,EXPDT,DATE_FMT
 CALL  CTSEDATE  USING FUNCTION,P_CDATE,CDATE DATE_FMT
 CALL  CTSEDATE  USING FUNCTION,P_OUTDATE,OUTDATE DATE_FMT
 SET P_CTIME = CTIME


 REPORT TITLE  @1 P_EXP_DAYNAM @11 ',' @13 P_EXP_MONNAM @24 P_EXP_DD
               @26 ',' @28 P_EXP_YYYY @32 '.' @33 P_EXP_DDD
               @50 COMPANY_NAME @115 'PAGE' @125 PAGENUMBER
        TITLE  @1 ' '
        TITLE  @1 'TMS REPORT-TMEOUTA  OUT OF AREA TAPES'

!
! SELECT ALL VOLUME RECORDS WHOSE OUTCODE IS NOT = 'VMS' AND NOT BLANK
!

 SELECT OUTCODE NOT = 'VMS' AND
        OUTCODE     > ' '

 CONTROL OUTCODE OUTDATE DSN CDATE CTIME VOLSEQ

 PRINT VOLSER DSN CJOB VOLSEQ P_CDATE
     P_CTIME P_EXPDT LABEL_TYPE TRTCH
     OUTCODE P_OUTDATE FLAG3_EDM

END
```



```
THURSDAY, JULY 21, 2005.200                      Y O U R   C O M P A N Y   N A M E                        PAGE      1

TMS REPORT-TMEOUTA  OUT OF AREA TAPES
--------------------------------------------------------------------------------------------------------------------------------
VOLSER DATA SET NAME                            CREATE      VOL CREATE       CREATE EXPIRATION LBL  TRTCH  LOC   DATE        EDM
                                                JOBNAME     SEQ DATE           TIME DATE       TYP         ID    MOVED       IND
--------------------------------------------------------------------------------------------------------------------------------
228511 AB.HOXDBM0A                              USRH0J0F     3  YYYY/191   06:34:00 YYYY/236   SL   18TK    NOR  YYYY/193    N
228275 AB.HOXDBM0A                              USRH0J0F     4  YYYY/191   06:34:00 YYYY/236   SL   18TK    NOR  YYYY/193    N
228351 AB.HOXDBM0A                              USRH0J0F     5  YYYY/191   06:34:00 YYYY/236   SL   18TK    NOR  YYYY/193    N
228520 AB.HOXDBM0A                              USRH0J0F     6  YYYY/191   06:34:00 YYYY/236   SL   18TK    NOR  YYYY/193    N
228327 AB.HOXFUM0A                              USRH0J0F     1  YYYY/191   05:35:00 YYYY/221   SL   18TK    NOR  YYYY/193    N
227457 AB.HOXHDM0A                              USRH0J1F     1  YYYY/191   07:22:00 YYYY/221   SL   18TK    NOR  YYYY/193    N
227376 AB.HOXHDM5A                              USRH0J26     1  YYYY/191   10:49:00 YYYY/221   SL   18TK    NOR  YYYY/193    N
224978 AB.HOXHDM5A                              USRH0J26     2  YYYY/191   10:49:00 YYYY/221   SL   18TK    NOR  YYYY/193    N
221945 AB.HOXHDM7A                              USRH0J27     1  YYYY/191   10:58:00 YYYY/221   SL   18TK    NOR  YYYY/193    N
227798 AB.HOXHDM7A                              USRH0J27     2  YYYY/191   10:58:00 YYYY/221   SL   18TK    NOR  YYYY/193    N
224320 AB.HOXJDM0G                              USRH0J11     1  YYYY/191   07:29:00 YYYY/221   SL   18TK    NOR  YYYY/193    N
224282 AB.HOXJKM0A                              USRH0J18     1  YYYY/191   10:43:00 YYYY/221   SL   18TK    NOR  YYYY/193    N
231592 AB.HOXKKM0A                              USRH0J18     1  YYYY/191   10:41:00 YYYY/221   SL   18TK    NOR  YYYY/193    N
223474 CA.YSRALM44                              YSRH0JM1     1  YYYY/194   00:11:00 YYYY/229   SL   18TK    NOR  YYYY/194    N
233447 CA.YSRBLX01.G0002V00                     YSRH0JCI     1  YYYY/193   19:26:00 YYYY/203   SL   18TK    NOR  YYYY/194    N
228355 CA.YSRCLX01.G0003V00                     YSRH0JM1     1  YYYY/194   00:04:00 YYYY/204   SL   18TK    NOR  YYYY/194    N
234201 CA.YSRCPX01                              YSRH0JM2     1  YYYY/194   00:29:00 YYYY/204   SL   18TK    NOR  YYYY/194    N
223045 CA.YSRGRX01.G0013V00                     YSRH0JM2     1  YYYY/194   00:54:00 YYYY/215   SL   18TK    NOR  YYYY/194    N
234456 CA.YSSDAX02.G0031V00                     YSSH0JDE     1  YYYY/193   17:20:00 YYYY/208   SL   18TK    NOR  YYYY/194    N
223781 CA.YSSGAX01                              YSSH0JDE     1  YYYY/193   17:15:00 YYYY/208   SL   18TK    NOR  YYYY/194    N
222776 CA.YSSHAX01.G0022V00                     YSSH0JDE     1  YYYY/193   17:27:00 YYYY/203   SL   18TK    NOR  YYYY/194    N
222812 CA.YSSJAX01.G0038V00                     YSSH0JDE     1  YYYY/193   17:28:00 YYYY/203   SL   18TK    NOR  YYYY/194    N
226559 DB.HOXJDX08                              USRH0J0D     1  YYYY/193   16:18:00 YYYY/214   SL   18TK    NOR  YYYY/194    N
222661 DB.HOXDBX01                              USRH0J0F     1  YYYY/154   09:55:00 YYYY/199   SL   18TK    NOR  YYYY/195    N
224155 DB.HOXDBX01                              USRH0J0F     2  YYYY/154   09:55:00 YYYY/199   SL   18TK    NOR  YYYY/195    N
234335 DB.HOXDBX01                              USRH0J0F     3  YYYY/154   09:55:00 YYYY/199   SL   18TK    NOR  YYYY/195    N
234216 DB.HOXDBX01                              USRH0J0F     4  YYYY/154   09:55:00 YYYY/199   SL   18TK    NOR  YYYY/195    N
236938 DB.HOXDBX01                              USRH0J0F     5  YYYY/154   09:55:00 YYYY/199   SL   18TK    NOR  YYYY/195    N
232902 DB.HOXDBX01                              USRH0J0F     6  YYYY/154   09:55:00 YYYY/199   SL   18TK    NOR  YYYY/195    N
228643 DB.HOXDBX01                              USRH0J0F     1  YYYY/156   05:39:00 YYYY/201   SL   18TK    NOR  YYYY/195    N
229913 DB.HOXDBX01                              USRH0J0F     2  YYYY/156   05:39:00 YYYY/201   SL   18TK    NOR  YYYY/195    N
229991 DB.HOXDBX01                              USRH0J0F     3  YYYY/156   05:39:00 YYYY/201   SL   18TK    NOR  YYYY/195    N
229919 DB.HOXDBX01                              USRH0J0F     4  YYYY/156   05:39:00 YYYY/201   SL   18TK    NOR  YYYY/195    N
228278 DB.HOXDBX01                              USRH0J0F     5  YYYY/156   05:39:00 YYYY/201   SL   18TK    NOR  YYYY/195    N
230373 DB.HOXDBX01                              USRH0J0F     6  YYYY/156   05:39:00 YYYY/201   SL   18TK    NOR  YYYY/195    N
237549 DB.HOXDBX01                              USRH0J0F     1  YYYY/181   04:52:00 YYYY/226   SL   18TK    NOR  YYYY/195    N
221842 DB.HOXDBX01                              USRH0J0F     2  YYYY/181   04:52:00 YYYY/226   SL   18TK    NOR  YYYY/195    N
236036 DB.HOXDBX01                              USRH0J0F     3  YYYY/181   04:52:00 YYYY/226   SL   18TK    NOR  YYYY/195    N
228016 DB.HOXDBX01                              USRH0J0F     4  YYYY/181   04:52:00 YYYY/226   SL   18TK    NOR  YYYY/195    N
220974 DB.HOXDBX01                              USRH0J0F     5  YYYY/181   04:52:00 YYYY/226   SL   18TK    NOR  YYYY/195    N
229666 DB.HOXDBX01                              USRH0J0F     6  YYYY/181   04:52:00 YYYY/226   SL   18TK    NOR  YYYY/195    N
225840 DB.HOXDBX01                              USRH0J7F     1  YYYY/182   23:08:00 YYYY/203   SL   18TK    NOR  YYYY/195    N
229726 DB.HOXDBX01                              USRH0J7F     2  YYYY/182   23:08:00 YYYY/203   SL   18TK    NOR  YYYY/195    N
225714 DB.HOXDBX01                              USRH0J7F     3  YYYY/182   23:08:00 YYYY/203   SL   18TK    NOR  YYYY/195    N
225513 DB.HOXDBX01                              USRH0J7F     4  YYYY/182   23:08:00 YYYY/203   SL   18TK    NOR  YYYY/195    N
225897 DB.HOXDBX01                              USRH0J7F     5  YYYY/182   23:08:00 YYYY/203   SL   18TK    NOR  YYYY/195    N
225878 DB.HOXDBX01                              USRH0J7F     6  YYYY/182   23:08:00 YYYY/203   SL   18TK    NOR  YYYY/195    N
220508 DB.HOXDBX01                              USRH0J0F     1  YYYY/195   04:38:00 YYYY/240   SL   18TK    NOR  YYYY/195    N
220552 DB.HOXDBX01                              USRH0J0F     2  YYYY/195   04:38:00 YYYY/240   SL   18TK    NOR  YYYY/195    N
221013 DB.HOXDBX01                              USRH0J0F     3  YYYY/195   04:38:00 YYYY/240   SL   18TK    NOR  YYYY/195    N
221090 DB.HOXDBX01                              USRH0J0F     4  YYYY/195   04:38:00 YYYY/240   SL   18TK    NOR  YYYY/195    N
233633 DB.HOXDBX01                              USRH0J0F     5  YYYY/195   04:38:00 YYYY/240   SL   18TK    NOR  YYYY/195    N
```

Example 3 - CA-Earl Out-of-Area Tapes

Example 4

This example illustrates how CA Earl can generate a scratch forecast report. A similar example resides as member TMESCRF in the CA 1 Source library.

```
!
!  TMS REPORT-TMESCRF   DATA SET SCRATCH FORECAST
!     ( SIMILAR TO OLD TMS REPORT 4 )
!         VOLUME RECORDS ONLY
!
 COPY EARLDEFS

 DEF FIRST_SW          (X 1) = ' '
 DEF P_SCRATCH_DATE   (X 10) = NONE 'SCRATCH' 'DATE'
 DEF SCRATCH_DATE    (P 7.0) = 0
 DEF EXP_DATA        (X 64)  = NONE
 DEF EXP_NAM                 = EXP_DATA 05-14 X
 DEF P_FROM_DATE      (X 10) = ' '
 DEF P_TO_DATE        (X 10) = ' '
 DEF WORK_DAYS       (P 7.0) = 7

 SET TYPRUN = R_VOLUME

!
!  CHANGE WORK_DAYS TO REFLECT THE DESIRED SCRATCH FORECAST NUMBER OF
!  DAYS. THIS WILL BE ADDED TO THE CURRENT DATE TO CREATE A DATE RANGE.
!
!  IF THE EXPIRATION DATE FALLS WITHIN THIS DATE RANGE, IT WILL BE
!  PRINTED ON THE SCRATCH FORECAST REPORT.
!

 GET TAPEDB
 GOTO EOJ TAPEDB = 'E'

 IF FIRST_SW = ' '
    THEN SET FIRST_SW = '*'
    SET FUNCTION = 'RETURN_DATE'
    CALL CTSEDATE USING FUNCTION,RUNDATE,RUNTIME
    SET FUNCTION = 'TO_EXPLODED'
    CALL CTSEDATE USING FUNCTION,P_EXP_DATA,RUNDATE
    SET DAYS = WORK_DAYS
    SET FUNCTION = 'ADD_DAYS'
    CALL CTSEDATE USING FUNCTION,RUNDATE,DAYS,SCRATCH_DATE
    SET FUNCTION = 'TO_EXTERNAL'
    CALL CTSEDATE  USING FUNCTION,P_FROM_DATE,RUNDATE,DATE_FMT
    CALL CTSEDATE  USING FUNCTION,P_TO_DATE,SCRATCH_DATE,DATE_FMT
 ENDIF

 SET FUNCTION = 'TO_EXPLODED'
 CALL CTSEDATE USING FUNCTION,EXP_DATA,EXPDT

 IF EXP_NAM NOT = 'JDATE' THEN
    GOTO START
 ENDIF
```

(continued on next page)

```
 SET    FUNCTION  = 'TO_EXTERNAL'
 CALL   CTSEDATE  USING FUNCTION,P_EXPDT,EXPDT,DATE_FMT
 CALL   CTSEDATE  USING FUNCTION,P_CDATE,CDATE,DATE_FMT
 CALL   CTSEDATE  USING FUNCTION,P_OUTDATE,OUTDATE,DATE_FMT

 SET P_CTIME = CTIME


 REPORT TITLE  @1 P_EXP_DAYNAM @11 ',' @13 P_EXP_MONNAM @24 P_EXP_DD
               @26 ',' @28 P_EXP_YYYY @32 '.' @33 P_EXP_DDD
               @50 COMPANY_NAME @115 'PAGE' @125 PAGENUMBER
        TITLE  @1 ' '
        TITLE  @1 'TMS REPORT-TMESCRF  JOBNAME SCRATCH FORECAST'
               @75 'REPORT RANGE IS' @91 P_FROM_DATE @102 'TO'
               @105 P_TO_DATE

!
! SELECT ALL VOLUME RECORDS WHOSE EXPIRATION DATE FALLS WITHIN THE
! CURRENT DATE AND THE FORECASTED END DATE (SCRATCH_DATE).
!

 SELECT EXPDT >= RUNDATE AND EXPDT <= SCRATCH_DATE

 CONTROL VOLSER

 PRINT VOLSER DSN 1 CJOB VOLSEQ P_CDATE P_CTIME P_EXPDT
       LABEL_TYPE TRTCH OUTCODE P_OUTDATE FLAG3_EDM

END
```

```
THURSDAY, JULY 21, 2005.200                      Y O U R   C O M P A N Y   N A M E                        PAGE      1

 TMS REPORT-TMESCRF  JOBNAME SCRATCH FORECAST                         REPORT RANGE IS YYYY/202   TO YYYY/209
 --------------------------------------------------------------------------------------------------------------------------------
 VOLSER DATA SET NAME                            CREATE      VOL CREATE       CREATE EXPIRATION LBL  TRTCH  LOC   DATE        EDM
                                                 JOBNAME     SEQ DATE           TIME DATE       TYP         ID    MOVED       IND
 --------------------------------------------------------------------------------------------------------------------------------
 121317 INTEXJ.D1019M82                          AXBACKUP     1  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 121640 INTEXJ.D1019M82                          AXBACKUP     1  YYYY/208   18:43:00 YYYY/208   SL   18TK   ????  ZEROS       N
 121886 INTEXJ.D1018M82                          AXBACKUP     1  YYYY/208   18:41:00 YYYY/208   SL   18TK   ????  ZEROS       N
 122598 INTEXJ.D10PWM92                          AXELJDMK     1  YYYY/202   05:57:00 YYYY/202   SL   18TK   ????  ZEROS       N
 123315 INTEXJ.D1001M50                          AXELJDMK     1  YYYY/206   21:57:00 YYYY/207   SL   18TK   ????  ZEROS       N
 123427 INTEXJ.D1001M85                          AXELJDMM    12  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 128265 INTEXJ.D1001M85                          AXELJDMM    10  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 128558 INTEXJ.D1001M85                          AXELJDMM     3  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 130372 INTEXJ.D1001M85                          AXELJDMM     1  YYYY/304   22:07:00 YYYY/209   SL   18TK   ????  ZEROS       N
 130575 INTEXJ.D1001M85                          AXELJDMM     5  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 130623 INTEXJ.D1001M85                          AXELJDMM     1  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 131238 INTEXJ.D10PWM92                          AXELJDMM     1  YYYY/209   01:05:00 YYYY/209   SL   18TK   ????  ZEROS       N
 131685 INTEXJ.D1001M85                          AXELJDMM     4  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 131810 INTEXJ.D1001M85                          AXELJDMM     9  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 132722 INTEXJ.D1001M85                          AXELJDMM     6  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 132762 INTEXJ.D1001M85                          AXELJDMM     2  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 134098 INTEXJ.D1018M82                          AXELJDMN     1  YYYY/208   18:43:00 YYYY/208   SL   18TK   ????  ZEROS       N
 134310 INTEXJ.D1019M81                          AXELJDMN     1  YYYY/208   18:42:00 YYYY/208   SL   18TK   ????  ZEROS       N
 134511 INTEXJ.D1001M85                          AXELJDMO    11  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 134881 INTEXJ.D1001M85                          AXELJDMO     8  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 134915 INTEXJ.D1001M85                          AXELJDMO     7  YYYY/299   09:49:00 YYYY/204   SL   18TK   ????  ZEROS       N
 139308 INTEXK.XXU19M81                          BORXBACK     1  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 139318 INTEXK.XXU19M81                          BORXBACK     2  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 139321 INTEXK.XXU19M81                          BORXBACK     3  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 139322 INTEXK.XXU19M81                          BORXBACK     4  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 139349 INTEXK.XXU19M81                          BORXBACK     5  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 139358 INTEXK.XXU19M81                          BORXBACK     6  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 139359 INTEXK.XXU19M81                          BORXBACK     7  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 139368 INTEXK.XXU19M81                          BORXBACK     8  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 139378 INTEXK.XXU19M81                          BORXBACK     9  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 139380 INTEXK.XXU19M81                          BORXBACK    10  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
 139381 INTEXK.XXU19M81                          BORXBACK    11  YYYY/208   18:40:00 YYYY/208   SL   18TK   ????  ZEROS       N
```

Example 4 - CA Earl Data Set Scratch Forecast

Example 5

In nonzero return code situations, CTSEDATE always includes an error message in your 80-byte CTSEDATE function code. This message contains the TMMDATE return code and reason code, and text which explains the failure. This example resides as member TMEDAT01 in the CA 1 Source library.

The following example shows one technique for handling nonzero return codes from CTSEDATE. The date format field is deliberately filled with the character X to cause an error. Note the additional coding and the CTSEDATE Error Report. You may want to use this code while debugging your CA Earl report. Once the process runs smoothly, you can delete the additional code and report output.

```
!
!  TMS REPORT-TMEDAT01   TAPE DATASETS EXPIRING IN JANUARY, YYYY
!                        CTSEDATE ERROR REPORT
!     ( SIMILAR TO OLD EARL07 REPORT )
!         VOLUME RECORDS ONLY
!

!
!     SAMPLE CA-EARL CODE TO DETECT AND PRINT ERRORS FROM CTSEDATE
!

 COPY EARLDEFS

 DEF CTSEDATE_RC    (B 5.0) = 00000 'CTSEDATE' 'RETURN CODE'
 DEF FIRST_SW         (X 1) = ' '

 IF FIRST_SW = ' '
    THEN SET FIRST_SW = '*'
    SET FUNCTION = 'RETURN_DATE'
    CALL CTSEDATE USING FUNCTION,RUNDATE,RUNTIME
    SET FUNCTION = 'TO_EXPLODED'
    CALL CTSEDATE USING FUNCTION,P_EXP_DATA,RUNDATE
 ENDIF

 SET FUNCTION = 'TO_EXTERNAL'
 SET DATE_FMT = 'XXXXXXXXXX'
 SET TYPRUN = R_VOLUME

!
! CHECK THE LAST RETURN CODE FROM CTSEDATE
!

 IF CTSEDATE_RC NOT = 0 THEN
    GOTO EOJ
 ENDIF

 GET TAPEDB
 GOTO EOJ TAPEDB = 'E'

 IF SCRATCH_IND NOT = 'N'
     GOTO START
 ENDIF

 CALL  CTSEDATE  USING FUNCTION,P_EXPDT,EXPDT,DATE_FMT
 SET CTSEDATE_RC = CTSEDATE
```

(continued from previous page)

```
 REPORT TITLE  @1 P_EXP_DAYNAM @11 ',' @13 P_EXP_MONNAM @24 P_EXP_DD
               @26 ',' @28 P_EXP_YYYY @32 '.' @33 P_EXP_DDD
               @50 COMPANY_NAME @115 'PAGE' @125 PAGENUMBER
        TITLE  @1 ' '
        TITLE  @1 'TMS REPORT-TMEDAT01'
                3 'TAPE DATASETS EXPIRING IN JANUARY, YYYY'
   SELECT EXPDT > 90000 AND EXPDT < 90032
   PRINT DSN VOLSER P_EXPDT

 REPORT TITLE  @1 P_EXP_DAYNAM @11 ',' @13 P_EXP_MONNAM @24 P_EXP_DD
               @26 ',' @28 P_EXP_YYYY @32 '.' @33 P_EXP_DDD
               @50 COMPANY_NAME @115 'PAGE' @125 PAGENUMBER
        TITLE  @1 ' '
        TITLE  @1 'TMS REPORT-TMEDAT01'
                3 'CTSEDATE ERROR REPORT'
   SELECT CTSEDATE_RC NOT = 0
   PRINT CTSEDATE_RC FUNCTION  EXPDT P_EXPDT DATE_FMT

END
```

```
THURSDAY, JULY 21, 2005.200                      Y O U R   C O M P A N Y   N A M E                        PAGE      1

 TMS REPORT-TMEDAT01   CTSEDATE ERROR REPORT
 -----------------------------------------------------------------------------------------------------------------------------
    CTSEDATE  FUNCTION                                                                            EXPIRATION   EXPIRATION   DATE_FMT
 RETURN CODE                                                                                            DATE   DATE
 -----------------------------------------------------------------------------------------------------------------------------
          8   TO_EXTERNAL    RC=8 RSN=32  BAD DATE FORMAT PARM                                      96/204                  XXXXXXXXXX

 END OF REPORT
```

CTSEDATE Error Report

Example 6

This example demonstrates how to use CA Earl to extract information from the CA 1 TMC to be used as input to one or more subsequent CA Earl jobs. This particular example extracts records which have a file sequence number greater than one. See Example 7 for using the extracted file. This example resides as member TMERPT01 in the CA 1 Source library.

```
!
!  TMS REPORT-TMERPT01   EXTRACT RECORDS USED WITH TMERPT02
!        VOLUME AND DSNB RECORDS
!

!
!  USE CA-EARL TO EXTRACT RECORDS FROM THE CA-1 TMC
!  AND CREATE A FILE CONTAINING ONLY THE EXTRACTED RECORDS.
!  THIS FILE WILL BE USED AS INPUT TO MEMBER TMERPT02 WITH
!  TYPRUN SET TO REPORT.
!
!  NOTES:
!     1)  ADD AN OUTFILE DD STATEMENT WITH LRECL=512 TO THE JCL.
!         THIS WILL CONTAIN THE EXTRACTED RECORDS AND BE THE
!         INPUT FOR AN EARL PROGRAM WITH TYPRUN = REPORT.
!     2)  MODIFY THE CA-EARL CONTROL STATEMENTS BASED ON YOUR
!         REQUIREMENTS.
!     3)  NO REPORT WILL BE PRODUCED BY THIS JOB SINCE THE OUTPUT
!         RECORDS ARE WRITTEN TO THE OUTPUT FILE.
!
!

 COPY EARLDEFS


 OUTFILE: FILE OUTPUT RECORD=512          ! DEFINE AN OUTPUT FILE
           DEF OUT_REC 1-512 X


 SET TYPRUN = R_SEQUENTIAL                ! LOOK AT ALL RECORDS

 GET TAPEDB

 GOTO EOJ TAPEDB ='E'

 IF FILESEQ > 1                           ! SELECT SECONDARY DATA SETS

     THEN
       SET OUT_REC = TAPE_DB_RECORD       ! COPY ALL DATA TO OUTPUT
       PUT OUTFILE                        ! AND WRITE IT
     ENDIF

END
```

Example 7

This example demonstrates how to use CA Earl to create a report using extracted information from a previous run of CA Earl. It is the same as Example 2 except the input is of extracted records only. See Example 6 for creating the extracted file. This example resides as member TMERPT02 in the CA 1 Source library.

```
!
!  TMS REPORT-TMERPT02   SELECTED VOLUME SERIAL MASTER REPORT
!        EXTRACTED RECORDS FROM TMERPT01
!

!
!  USE CA-EARL TO GENERATE THE VOLUME SERIAL MASTER RPT.
!           USE EXTRACTED RECORDS ONLY.
!
!  NOTES:
!     1)  SEE MEMBER TMERPT01 FOR A SAMPLE EXTRACT PROGRAM.
!     2)  IN THE JCL, TAPEDB SHOULD POINT TO THE EXTRACT FILE.
!     3)  MODIFY THE CA-EARL CONTROL STATEMENTS BASED ON YOUR
!         REQUIREMENTS.
!

 COPY EARLDEFS

 DEF FIRST_SW          (X 1) = ' '


 SET TYPRUN = R_REPORT

 GET TAPEDB
 GOTO EOJ TAPEDB = 'E'

 IF SCRATCH_IND = 'N' OR ACTIND = 'Y' THEN
     GOTO CONTINUE
   ELSE
     GOTO START
 ENDIF

 CONTINUE:
 IF FIRST_SW = ' '
    THEN SET FIRST_SW = '*'
    SET FUNCTION = 'RETURN_DATE'
    CALL CTSEDATE USING FUNCTION,RUNDATE,RUNTIME
    SET FUNCTION = 'TO_EXPLODED'
    CALL CTSEDATE USING FUNCTION,P_EXP_DATA,RUNDATE
 ENDIF

 SET FUNCTION = 'TO_EXTERNAL'
 CALL  CTSEDATE  USING FUNCTION,P_EXPDT,EXPDT,DATE_FMT
 CALL  CTSEDATE  USING FUNCTION,P_CDATE,CDATE DATE_FMT
 CALL  CTSEDATE  USING FUNCTION,P_LDATE,LDATE,DATE_FMT
```

(continued on next page)

```
 SET P_CTIME = CTIME

 REPORT TITLE  @1 P_EXP_DAYNAM @11 ',' @13 P_EXP_MONNAM @24 P_EXP_DD
               @26 ',' @28 P_EXP_YYYY @32 '.' @33 P_EXP_DDD
               @50 COMPANY_NAME @115 'PAGE' @125 PAGENUMBER
        TITLE  @1 ' '
        TITLE  @1 'TMS REPORT-TMERPT02'
                3 'SELECTED VOLUME SERIAL MASTER REPORT'

 CONTROL VOLSER FILESEQ

 PRINT VOLSER FILESEQ DSN CJOB VOLSEQ P_CDATE
     P_CTIME P_EXPDT P_LDATE LABEL_TYPE FLAG3_EDM

END
FRIDAY, JULY 21, 2005.201                      Y O U R   C O M P A N Y   N A M E                        PAGE      1

TMS REPORT-TMERPT02   SELECTED VOLUME SERIAL MASTER REPORT
------------------------------------------------------------------------------------------------------------------------------
VOLSER   FILE DATA SET NAME                             CREATE       VOL  CREATE        CREATE  EXPIRATION  LAST USE  LBL  EDM
          SEQ                                           JOBNAME      SEQ  DATE            TIME  DATE        DATE      TYP  IND
------------------------------------------------------------------------------------------------------------------------------
000003     2  MVMF.V000003.FILE2                        CAICA11C      0   YYYY/100    07:06:00  YYYY/300    ZEROS
000003     3  MVMF.V000018.FILE3                        CAICA11C      0   YYYY/100    07:06:00  YYYY/300    ZEROS
000003     4  MVMF.V000018.FILE4                        CAICA11C      0   YYYY/100    07:06:00  YYYY/300    ZEROS
000003     5  MVMF.V000008.FILE5                        CAICA11C      0   YYYY/100    07:06:00  YYYY/300    ZEROS
000006     2  MVMF.V000006.FILE2                        CAICA11B      0   YYYY/105    07:06:00  YYYY/300    ZEROS
000006     3  MVMF.V000006.FILE3                        CAICA11B      0   YYYY/105    07:06:00  YYYY/300    ZEROS
000006     4  MVMF.V000010.FILE4                        CAICA11B      0   YYYY/105    07:06:00  YYYY/300    ZEROS
000006     5  MVMF.V000010.FILE5                        CAICA11B      0   YYYY/105    07:06:00  YYYY/300    ZEROS
000009     2  MVMF.V000009.FILE2                        CAICA11A      0   YYYY/102    07:06:00  YYYY/300    ZEROS
000009     3  MVMF.V000009.FILE3                        CAICA11A      0   YYYY/102    07:06:00  YYYY/300    ZEROS
000009     4  MVMF.V000011.FILE4                        CAICA11A      0   YYYY/102    07:06:00  YYYY/300    ZEROS
000009     5  MVMF.V000011.FILE5                        CAICA11A      0   YYYY/102    07:06:00  YYYY/300    ZEROS
000015     2  MF.V000015.FILE2                          CAICA11A      0   YYYY/100    07:06:00  CATALOG     ZEROS
000015     3  MF.V000015.FILE3                          CAICA11A      0   YYYY/100    07:06:00  CATALOG     ZEROS
000015     4  MF.V000015.FILE4                          CAICA11A      0   YYYY/100    07:06:00  YYYY/300    ZEROS
000019     2  MF.V000019.FILE2                          CAICA11B      0   YYYY/100    07:06:00  CATALOG     ZEROS
000019     3  MF.V000019.FILE3                          CAICA11B      0   YYYY/100    07:06:00  CATALOG     ZEROS
000019     4  MF.V000019.FILE4                          CAICA11B      0   YYYY/100    07:06:00  CATALOG     ZEROS
000023     2  MVMF.V000023.FILE2                        CAICA11C      0   YYYY/100    07:06:00  YYYY/300    ZEROS
000023     3  MVMF.V000038.FILE3                        CAICA11C      0   YYYY/100    07:06:00  YYYY/300    ZEROS
000023     4  MVMF.V000038.FILE4                        CAICA11C      0   YYYY/100    07:06:00  YYYY/300    ZEROS
000023     5  MVMF.V000028.FILE5                        CAICA11C      0   YYYY/100    07:06:00  YYYY/300    ZEROS
000026     2  MVMF.V000026.FILE2                        CAICA11B      0   YYYY/105    07:06:00  YYYY/300    ZEROS
000026     3  MVMF.V000026.FILE3                        CAICA11B      0   YYYY/105    07:06:00  YYYY/300    ZEROS
000026     4  MVMF.V000030.FILE4                        CAICA11B      0   YYYY/105    07:06:00  YYYY/300    ZEROS
000026     5  MVMF.V000030.FILE5                        CAICA11B      0   YYYY/105    07:06:00  YYYY/300    ZEROS
000029     2  MVMF.V000029.FILE2                        CAICA11A      0   YYYY/102    07:06:00  YYYY/300    ZEROS
000029     3  MVMF.V000029.FILE3                        CAICA11A      0   YYYY/102    07:06:00  YYYY/300    ZEROS
000029     4  MVMF.V000031.FILE4                        CAICA11A      0   YYYY/102    07:06:00  YYYY/300    ZEROS
000029     5  MVMF.V000031.FILE5                        CAICA11A      0   YYYY/102    07:06:00  YYYY/300    ZEROS

END OF REPORT
```

Example 7 - CA Earl Selected Volume Serial Master Report

Example 8

Member CTETEVLT of the CA 1 or CA TLMS EARL data set hlq.CTAPEARL is for clients who also run the CA Tape Encryption product to identify if any tapes being sent off-site are not protected with encryption. CA Tape Encryption creates a unique BES Key Index for each new symmetric key instance. The BES Key Index is saved in the CA 1, CA TLMS, or IBM DFSMSrmm tape catalog record for each file that CA Tape Encryption encrypts through direct integration between the products. The EARL field BES_KEY, which is located in the common record, reports on the BES Key Index. If the BES_KEY field is not zero, the file is encrypted.

Report CTETEVLT uses this technique and checks the BES_KEY for each file in the tape catalog that is selected for movement off-site "today." If any of these are found to have a BES_KEY of zero, the report ends with an RC=8 to alert you that you may not be protecting all tapes that you intended.

Report CTETEVLT uses the new CTSEPM pattern masking capability to either include or exclude data sets matching specific name patterns. The pattern matching techniques used in the CTETEVLT EARL report provide a good example for exploiting this feature in user written EARL reports.

Appendix A: Report Cross-Reference

This appendix contains two report cross-references, one by TMS report number (when applicable) and the other by the generating program name. For r11 and above reports do not have a TMS report number.

Introduction

If the generating program listed for Version 5.1 or higher was known by a different name in Version 5.0, that program name is included in parentheses. For example, TMEJOB01 (TMSRPT1) is known as TMEJOB01 in Version 5.1 and higher and was known as TMSRPT1 in Version 5.0.

The generating programs that have a name beginning with TME are CA Earl report members which are provided in the CA 1 Source library.

BATCHIDs

The batch ID contained in the TMC record identifies the last batch program that updated the record. This includes CA 1 batch and online programs, and user defined programs that have been assigned a unique batchid. A complete list of defined batchids can be found in the source module TMSKEYAB. TMMKEY macro entries specifying KEYL=BATCHID list the batchid and the corresponding program name.

Report Number Cross-Reference

Report Number	Report Name	Generating Program
TMS Report 00	Initialization Report	TMSINIT
TMS Report 01	Volume Serial Master	TMEVSNM (TMSRPT4)
TMS Report 02	Jobname Sequence	TMEJOB01 (TMSRPT1)
TMS Report 03	Jobname Scratch Forecast	TMEJOB02 (TMSRPT1)
TMS Report 04	Data set Scratch Forecast	TMESCRF (TMSRPT3)
TMS Report 05	Scratch and Clean Listing for SL/NL Tape Volumes	TMECLN01
TMS Report 06	Scratch and Clean Relist	TMECLN04

Report Number	Report Name	Generating Program
TMS Report 07	Out-of-Area Tapes	TMEOUTA (TMSRPT5)
TMS Report 08	High Retention Tapes	TMERETN (TMSRPT6)
TMS Report 09	Expired Catalog Listing	TMECTLG
TMS Report 10	Data Set Name Sequence	TMEDSN01 (TMSRPT2)
TMS Report 11	Data Set Name Scratch Forecast	TMEDSN02 (TMSRPT2)
TMS Report 12	Batch Inquiry (Headers)	TMSBINQ
TMS Report 12	Batch Inquiry (Long)	TMSBINQ
TMS Report 12	Batch Inquiry (Short)	TMSBINQ
TMS Report 12	Batch Inquiry (Dump)	TMSBINQ
TMS Report 13	List Available TMC Records	TMSDELET
TMS Report 13	Add (Logically) TMC Records	TMSDELET
TMS Report 13	Delete (Logically) TMC Records	TMSDELET
TMS Report 14	TMC Batch Updates	TMSUPDTE
TMS Report 14	TMSUPDTE Error Cross-Reference Report	TMSUPDTE
TMS Report 15	Invalid Date and Packed Fields	TMSCKDTM
TMS Report 16	Expired Cycle Control Listing	TMECYCLE
TMS Report 17	Tape Management Catalog and OS Catalog Differences	TMSOSCAT
TMS Report 18	User-Supplied Input Converted to TMRECORD Format User-Supplied Input Converted to TMEXPDT Format User-Supplied Input Converted to TMSUPDTE Format	TMSCONVR

Report Number	Report Name	Generating Program
TMS Report 19	Multivolume/Multi-Data Set List	TMEMVMD (TMSRPT7)
TMS Report 20	Addition and Deletion of Data Sets	TMSAGGR
TMS Report 22	CA 1 Batch Expiration Update Listing	TMEXPDT
TMS Report 22	CA 1 Default Expiration Update Listing	TMEXPDT
TMS Report 23	Vault Pattern Data Set	TMSVMEDT
TMS Report 24	Vault Management System - Exceptional Conditions Encountered During Processing	TMSVMVLT
TMS Report 25	Picking List for Vault	TMEVLT01 (TMSVMLST)
TMS Report 25	Distribution List for Vault	TMEVLT02 (TMSVMLST)
TMS Report 25	Inventory List for Vault	TMEVLT03 (TMSVMLST)
TMS Report 26	CA 1 Vault Update Listing	TMSVMUPD
TMS Report 27	Extend Expiration Dates	TMECLN03
TMS Report 29	Invalid Pointers	TMSPTRS
TMS Report 30	TMSFORMT Audit Trail	TMSFORMT
TMS Report 31	Backup/Restore the TMC File	TMSCOPY
TMS Report 32	Audit Data Set Activity	TMSAUDIT
TMS Report 32	Batch Update Audit Records (Type 0)	TMSAUDIT
TMS Report 32	Input Activity Audit Records (Type 1)	TMSAUDIT
TMS Report 32	Output Activity Audit Records (Type 2)	TMSAUDIT
TMS Report 32	Audit Exception Records (Type 3)	TMSAUDIT
TMS Report 32	Audit NOT SCRATCH Rejects (Type 4)	TMSAUDIT

Report Number	Report Name	Generating Program
TMS Report 32	Audit of Functional Modifications (Type 5)	TMSAUDIT
TMS Report 33	Volume Chain Generation Report	TMSMULV
TMS Report 34	Volume Chain Error Report	TMSMULV
TMS Report 35	Build TMC Records From Catalog Listings	TMSIDATA
TMS Report 36	TMS Sequential Data Sets	TMSDATA
TMS Report 37	Pull List	TMSPULL
TMS Report 38	Setup List	TMSPULL
TMS Report 39	Purged TMC Records	TMSREMOV
TMS Report 40	Purge Errors	TMSREMOV
TMS Report 41	Tape Initialization	TMSTPNIT
TMS Report 42	Scratch and Clean Reset	TMECLN02
TMS Report 43	Conversion Exception Report	TMSCONTM
TMS Report 44	RDS Conversion Error Report	TMSCONRD
TMS Report 45	Vault Pattern Conversion Error Report	TMSCONVP
TMS Report 46	Conversion Activity Report	TMSCONTM
TMS Report 47	TMSC2TMC Exception Report	TMSC2TMC
TMS Report 48	TMC DSNB and CTL Updates	TMSUDSNB
TMS Report 50	TMC/DSNB Records for Export	TMSSPLIT
TMS Report 51	TMC/DSNB Records for Import	TMSMERGE
TMS Report 52	TMC Volumes for Export	TMSSPLIT

Report Number	Report Name	Generating Program
TMS Report 53	TMSINITD Batch Report	TMSINITD
TMS Report 54	Options Table Status Report	TMSSTATS
TMS Report 55	CA 1 Message Intercept Table Entries Currently Active	TMSSTATS
TMS Report 56	Scratch Pool Definition Report	TMSSTATS
TMS Report 57	Scratch Pool Assignment Report	TMSSTATS
TMS Report 58	Error Report for TMSRPI	TMSRPI
TMS Report 59	DSNB Initialization Report	TMSFVSN
TMS Report 61	EDM Rule Assignment Report	TMSSTATS
TMS Report 62	Tapes to be Ejected	TMEEJECT
TMS Report 63	Virtual Volume Inventory	TMEVTINV
TMS Report 64	Tapes to be Imported	TMEIMPRT
TMS Report 65	TMSC2TMC Totals Report	TMSC2TMC
TMS Report 67	Keyword Table Report	TMSSTATS
TMS Report 68	Security Table Report	TMSSTATS
TMS Report 70	Tape Management Database Synchronization	DYNSYNC
TMS Report 71	CA 1 CA Dynam/T Interface Update Report	DYNUPDTE
TMS Report 72	CA 1 CA Dynam/T Interface Exception Report	DYNUPDTE
TMS Report 73	CA 1 CA Dynam/T Interface Queue Format Report	DYNFORMT

Report Number	Report Name	Generating Program
TMS Report 74	Check Levels of CA 1 LPA and Link List Modules	TMSCKLVL
TMS Report 80	TMEAUDEX Messages	TMEAUDEX
TMS Report 81	TMSBLDVR	TMSBLDVR
TMS Report 82	TMEXPDT Messages	TMEXPDT
TMS Report 82	Retention Data Set	TMEXPDT
TMS Report 83	TMSCTLG Messages	TMSCTLG
TMS Report 84	TMSCYCLE Messages	TMSCYCLE
TMS Report 85	TMSCLEAN Messages	TMSCLEAN
TMS Report 86	TMSCLNOA Messages	TMSCLNOA
TMS Report 88	TMSRULD Messages	TMSRULD
	SMS Update Listing	TMSSMSUP
	Test Tape OPENs	CTSTAPER
	Detail Log File	DYNCOMM
	Error Log File	DYNCOMM
	Scratch and Clean RESETCAT	TMECLN05
	Volume Serial Number Master	TMEVSNM (TMSRPT4)
	Uncatalog Tape Data Sets	TMSUNCAT
TMS Report 89	TMSCLEAN RANGE REPORT	TMSCLEAN .row
TMS Report CTETEVLT	For CA Tape Encryption customers - Report volumes with BESKEY=0 that are selected to move to offsite vault.	EARL (CTETEVLT)
TMS Report CTE3495	Synchronize IBM Tape Robot to Tape System	EARL (CTE3495)
TMS Report TMEAUD01	Audit Records Type=	EARL (TMEAUD01)
TMS Report TMEAUD02	Total Audit Listed Type 0,1,2,3,4 Volume Range	EARL (TMEAUD02)

Report Number	Report Name	Generating Program
TMS Report TMEAUD03	Audit Records Type=	EARL (TMEAUD03)
TMS Report TMEDAT01	Tape Data Sets Expiring in xxxx	EARL (TMEDAT01)
TMS Report TMEDAT02	Tapes Used Between Two Dates	EARL (TMEDAT02)
TMS Report TMEFLAG	Volume Record Flags and DSNB Record Flags	EARL (TMEFLAG)
TMS Report TMERPT01	Extract Records Used with TMERPT02 Volume and DSNB Records	EARL (TMERPT01)
TMS Report TMERPT02	Selected Volume Serial Master Report Extracted Records from TMERPT01	EARL (TMERPT02)
TMS Report TMESMS	SMS Update Listing Volume and DSNB Records	EARL (TMERPT02)

Program to Report Cross-Reference

Generating Program	Report Number	Report Name
CTETEVLT	CTETEVLT	For CA Tape Encryption customers - Report volumes with BESKEY=0 that are selected to move to offsite vault.
CTE3495	CTE3495	Synchronize IBM Tape Robot to Tape System
CTSTAPER	-	Test Tape OPENs
DYNCOMM	-	Detail Log File
DYNCOMM	-	Error Log File
DYNSYNC	TMS Report 70	Tape Management Database Synchronization
DYNUPDTE	TMS Report 71	CA 1 CA Dynam/T Interface Update
DYNUPDTE	TMS Report 72	CA 1 CA Dynam/T Interface Exception Report

Generating Program	Report Number	Report Name
DYNFORMT	TMS Report 73	CA 1 CA Dynam/T Interface Queue Format
TMEAUD01	TMS Report TMEAUD01	Audit Records Type=
TMEAUD02	TMS Report TMEAUD02	Total Audit Listed Type 0,1,2,3,4 Volume Range
TMEAUD03	TMS Report TMEAUD03	Audit Records Type=
TMECLN01	TMS Report 05	Scratch and Clean Listing for NL/SL Tape Volumes
TMECLN02	TMS Report 42	Scratch and Clean Reset
TMECLN03	TMS Report 27	Extend Expiration Dates
TMECLN04	TMS Report 06	Scratch and Clean Relist
TMECLN05		Scratch and Clean RESETCAT
TMECLNOA	TMS Report 87	Scratch and Clean for NL/SL Tape Volumes
TMECTLG	TMS Report 09	Expired Catalog Listing
TMECYCLE	TMS Report 16	Expired Cycle Control Listing
TMEDAT01	TMS Report TMEDAT01	Tape Data Sets Expiring in xxxx
TMEDAT02	TMS Report TMEDAT02	Tapes Used Between Two Dates
TMEDSN01 (TMSRPT2)	TMS Report 10	Data Set Name Sequence
TMEDSN02 (TMSRPT2)	TMS Report 11	Data Set Name Scratch Forecast
TMEFLAG	TMS Report TMEFLAG	Volume Record Flags and DSNB Record Flags
TMEJOB01 (TMSRPT1)	TMS Report 02	Jobname Sequence
TMEJOB02 (TMSRPT1)	TMS Report 03	Jobname Scratch Forecast
TMEMVMD (TMSRPT7)	TMS Report 19	Multivolume/Multi-Data Set List

Generating Program	Report Number	Report Name
TMEOUTA (TMSRPT5)	TMS Report 07	Out-of-Area Tapes
TMERETN (TMSRPT6)	TMS Report 08	High Retention Tapes
TMERPT01	TMS Report TMERPT01	Extract Records Used with TMERPT02 Volume and DSNB Records
TMERPT02	TMS Report TMERPT02	Selected Volume Serial Master Report Extracted Records from TMERPT01
TMESCRF (TMSRPT3)	TMS Report 04	Data Set Scratch Forecast
TMESMS	TMS Report TMESMS	SMS Update Listing Volume and DSNB Records
TMEVLT01 (TMSVMLST)	TMS Report 25	Picking List for Vault
TMEVLT02 (TMSVMLST)	TMS Report 25	Distribution List for Vault
TMEVLT03 (TMSVMLST)	TMS Report 25	Inventory List for Vault
TMEVSNM (TMSRPT4)	TMS Report 1	Volume Serial Number Master
TMEXPDT	TMS Report 22	CA 1 Batch Expiration Update Listing
TMEXPDT	TMS Report 22	CA 1 Default Expiration Date Listing
TMEXPDT	TMS Report 82	TMEXPDT Messages
TMEXPDT	TMS Report 82	Retention Data Set
TMSAGGR	TMS Report 20	Addition and Deletion of Data Sets
TMSAUDEX	TMS Report 80	TMEAUDEX Messages
TMSAUDIT	TMS Report 32	Audit Data Set Activity
TMSAUDIT	TMS Report 32	Batch Update Audit Records (Type 0)
TMSAUDIT	TMS Report 32	Input Activity Audit Records (Type 1)
TMSAUDIT	TMS Report 32	Output Activity Audit Records (Type 2)
TMSAUDIT	TMS Report 32	Audit Exception Records (Type 3)
TMSAUDIT	TMS Report 32	Audit NOT SCRATCH Rejects (Type 4)

Generating Program	Report Number	Report Name
TMSAUDIT	TMS Report 32	Audit of Functional Modifications (Type 5)
TMSBINQ	TMS Report 12	Batch Inquiry (Headers)
TMSBINQ	TMS Report 12	Batch Inquiry (Long)
TMSBINQ	TMS Report 12	Batch Inquiry (Short)
TMSBINQ	TMS Report 12	Batch Inquiry (Dump)
TMSBLDVR	TMS Report 81	TMSBLDVR Messages
TMSCKDTM	TMS Report 15	Invalid Date and Packed Fields
TMSCKLVL	TMS Report 74	Check Levels of CA 1 LPA and Link List Modules
TMSCLEAN	TMS Report 85	TMSCLEAN Messages
TMSCLNOA	TMS Report 86	TMSCLNOA Messages
TMSCONRD	TMS Report 44	RDS Conversion Error Report
TMSCONTM	TMS Report 43	Conversion Exception Report
TMSCONTM	TMS Report 46	Conversion Activity Report
TMSCONVP	TMS Report 45	Vault Pattern Conversion Error Report
TMSCONVR	TMS Report 18	User-Supplied Input Converted to TMRECORD Format
TMSCONVR	TMS Report 18	User-Supplied Input Converted to TMEXPDT Format
TMSCONVR	TMS Report 18	User-Supplied Input Converted to TMSUPDTE Format
TMSCOPY	TMS Report 31	Backup/Restore the TMC File
TMSCTLG	TMS Report 83	TMSCTLG Messages
TMSCYCLE	TMS Report 84	TMSCYCLE Messages
TMSDATA	TMS Report 36	TMS Sequential Data Sets
TMSDELET	TMS Report 13	List Available TMC Records
TMSDELET	TMS Report 13	Add (Logically) TMC Records
TMSDELET	TMS Report 13	Delete (Logically) TMC Records
TMEEJECT	TMS Report 62	Tapes to be Ejected
TMEIMPRT	TMS Report 63	Tapes to be Imported

Generating Program	Report Number	Report Name
TMEVTINV	TMS Report 64	Virtual Volume Inventory
TMSFORMT	TMS Report 30	TMSFORMT Audit Trail
TMSFVSN	TMS Report 59	DSNB Initialization
TMSIDATA	TMS Report 35	Build TMC Records From Catalog Listings
TMSINIT	TMS Report 00	Initialization Report
TMSINITD	TMS Report 53	TMSINITD Batch Report
TMSMERGE	TMS Report 51	TMC/DSNB Records for Import
TMSMULV	TMS Report 33	Volume Chain Generation Report
TMSMULV	TMS Report 34	Volume Chain Error Report
TMSOSCAT	TMS Report 17	Tape Management Catalog and OS Catalog Differences
TMSPTRS	TMS Report 29	Invalid Pointers
TMSPULL	TMS Report 37	Pull List
TMSPULL	TMS Report 38	Setup List
TMSREMOV	TMS Report 39	Purged TMC Records
TMSREMOV	TMS Report 40	Purge Errors
TMSRPI	TMS Report 58	Error Report for TMSRPI
TMSRULD	TMS Report 88	TMSRULD Messages
TMSSMSUP	TMS Report 87	SMS Update Listing
TMSSPLIT	TMS Report 50	TMC/DSNB Records for Export
TMSSPLIT	TMS Report 52	TMC Volumes for Export
TMSSTATS	TMS Report 54	Options Table Status Report
TMSSTATS	TMS Report 55	CA 1 Message Intercept Table Entries Currently Active
TMSSTATS	TMS Report 56	Scratch Pool Definition Report
TMSSTATS	TMS Report 57	Scratch Pool Assignment Report
TMSSTATS	TMS Report 61	EDM Rule Assignment Report
TMSTPNIT	TMS Report 41	Tape Initialization
TMSUDSNB	TMS Report 48	TMC DSNB and CTL Updates
TMSUNCAT		Uncatalog Tape Data Sets

Generating Program	Report Number	Report Name
TMSUPDTE	TMS Report 14	TMSUPDTE Error Cross-Reference Report
TMSUPDTE	TMS Report 14	TMC Batch Updates
TMSVMEDT	TMS Report 23	Vault Pattern Data Set
TMSVMUPD	TMS Report 26	CA 1 Vault Update Listing
TMSVMVLT	TMS Report 24	Vault Management System - Exceptional Conditions Encountered During Processing
TMSCLEAN	TMS Report 89	TMSCLEAN Range Report

The following is a sample report produced by a TMSCLEAN RANGE operation:

```
THURSDAY , JUNE 09, 2008.158   Y O U R  C O M P A N Y  N A M E                                                       PAGE 1

TMS REPORT-89         TMSCLEAN RANGE REPORT

CONTROL STATEMENT IN EFFECT:
-----------------------------------------------------------------------
VOL=A00000,A00005
VOL=CA3000 CA4000
SCRPOOL=SPOOL2
FILE1DSN=THIS.IS.*.TEST.FILE
FILE1DSN=THIS.IS.NOT.TEST.**
```

Index