

DFSMS/MVS Version 1 Release 5



Using Magnetic Tapes

DFSMS/MVS Version 1 Release 5



Using Magnetic Tapes

Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page xi.

Third Edition (March 1999)

This edition applies to Version 1 Release 5 of DFSMS/MVS (5695-DF1), Release 7 of OS/390 (5647-A01), and any subsequent releases until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

A form for readers' comments appears at the back of this publication. If the form has been removed, address your comments to:

International Business Machines Corporation
RCF Processing Department
G26/050
5600 Cottle Road
SAN JOSE, CA 95193-0001
U.S.A.

Or you can send your comments electronically to starpubs@vnet.ibm.com.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© **Copyright International Business Machines Corporation 1972,1999. All rights reserved.**

Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	xi
Programming Interface Information	xi
Trademarks	xii
 About This Book	 xiii
Required Product Knowledge	xiii
Required Publications	xiii
How to Tell if this Book is Current	xiv
Related Publications	xiv
Referenced Publications	xiv
References to Product Names Used in DFSMS/MVS Publications	xv
 Summary of Changes	 xvii
Third Edition, March 1999	xvii
 Chapter 1. Introduction to Tape Processing	 1
Describing the Labels	2
Describing the Data Sets	3
Allocating Tape Data Sets	3
Completing the Data Control Block	3
Data Sets Managed by Storage Management Subsystem	5
Data Sets and Volumes Managed by DFSMSrmm	5
Cataloged Data Sets	5
Generation Data Groups	5
Concatenated Data Sets	7
Passed Data Sets	7
Multiple Volumes and Multiple Data Sets	7
Processing Methods and Routines	8
Open/Close/End-of-Volume Routines	9
Checkpoint/Restart Routine	9
Automatic Volume Recognition	10
Tape Disposition	10
Verifying Block Count	11
Bypassing Label Processing	11
Tape Characteristics	12
Nine-Track Tapes and Dual Density Feature	13
Seven-Track Tapes	13
Tape Cartridges	15
Tapemarks	15
Beginning and End of Tape	15
 Chapter 2. IBM Standard Labels	 17
Label Definitions and Organization	17
Volume Label	20
Data Set Header Labels	20
User Header Labels	20
Data Set Trailer Labels	20
User Trailer Labels	21
Additional Labels	21
System-Managed Mountable Tape External Labels	21

Tapemarks	21
IBM Standard Labels Format	22
Format of the IBM Standard Volume Label (VOL1)	22
Format of IBM Standard Data Set Label 1 (HDR1/EOV1/EOF1)	25
Format of IBM Standard Data Set Label 2 (HDR2/EOV2/EOF2)	32
Format of IBM Standard User Label (UHL1-UHL8 or UTL1-UTL8)	38
IBM Standard Label Processing	40
Creating a Volume Label	41
IBM Standard Volume Label Verification and Editor Routines	43
Protecting Data	43
Opening an Input Data Set	45
Volume Serial Number	45
Positioning the Volume to the Data Set	45
Processing the HDR1 Label	47
Processing the HDR2 Label	48
User Header Labels	49
Read Backward	49
End-of-Data or End-of-Volume on Input	49
Verifying Block Count in Label When Reading	50
The EOVS/EOF2 Label	50
User Trailer Labels	50
Determining Volume Switch	50
Checking the Next Volume	52
Closing an Input Data Set	53
Opening an Output Data Set	53
Volume Serial Number	54
Positioning the Volume to the Data Set	55
Open/EOV User Exit for Security Verification	56
Expiration Date on Existing Label	57
Password Protection and Data Set Name on Existing Label	57
Writing Data Set Header Labels	58
Writing User Header Labels	58
Permanent I/O Error	58
End-of-Volume on Output	58
Writing Data Set Trailer Labels	59
Writing User Trailer Labels	59
Writing Labels on New Volume	59
RACF Processing on the New Volume	60
Special End-of-Volume Conditions	60
Closing an Output Data Set	60
Writing Data Set Trailer Labels	60
Writing User Trailer Labels	61
Restarting from a Checkpoint	61
 Chapter 3. ISO/ANSI Labels	 63
Label Definition and Organization	64
Volume Label	68
Data Set Header Labels	68
User Header Labels	69
Data Set Trailer Labels	69
User Trailer Labels	70
System-Managed Mountable Tape External Labels	70
Tapemarks	70
ISO/ANSI Labels Format	71

	Format of the ISO/ANSI Volume Label (VOL1)	71
	Format of ISO/ANSI Data Set Label 1 (HDR1/EOV1/EOF1)	74
	Format of ISO/ANSI Data Set Label 2 (HDR2/EOV2/EOF2)	81
	Format of ISO/ANSI User Header and Trailer Labels (UHL/UTL)	87
	Other File Header Labels	88
	Overview of ISO/ANSI Label Processing	90
	Creating a Volume Label	92
	ISO/ANSI Volume Label Verification and Editor Routines	94
	Protecting Data	94
	Volume Accessibility	96
	Data Set Accessibility	97
	ISO/ANSI Installation and Use Characteristics	99
	ASCII Conversion	99
	CCSID Conversion	99
	Volume Initialization	99
	ISO/ANSI Installation Exits	99
	RACHECK Installation Exits	101
	Generation Data Groups	101
	Spanned Records	101
	Block Padding	101
	Checkpoints	101
	Compatibility with Non-Version 3 or Non-Version 4 Volumes	102
	Processing Differences Between Version 1 and Version 3 or Version 4	102
	Processing Version 3 or Version 4 Volumes on Earlier MVS Systems	102
	Tapemarks	103
	Opening an Input Data Set	103
	Volume Serial Number	104
	Positioning the Volume to the Data Set	104
	Processing the HDR1 Label	106
	Processing the HDR2 Label	107
	Processing User Header Labels	108
	Read Backward	108
	End-of-Data or End-of-Volume on Input	108
	Verifying Block Count in Label When Reading	109
	Determining Volume Switch	109
	Checking the Next Volume	111
	Closing an Input Data Set	111
	Opening an Output Data Set	112
	Volume Serial Number	113
	Positioning the Volume to the Data Set	113
	Expiration Date on Existing Label	114
	Writing Data Set Header Labels	114
	Writing User Header Labels	115
	Permanent I/O Error	115
	EOV on Output	115
	Writing Data Set Trailer Labels	116
	Writing User Trailer Labels	116
	Labels on New Volume	116
	RACF Processing on the New Volume	116
	Special EOV Conditions	117
	Closing an Output Data Set	117
	Writing Data Set Trailer Labels	117
	Writing User Trailer Labels	117
	Checkpoint/Restart Not Allowed	118

Chapter 4. Nonstandard Labels	119
Nonstandard Volume Label Verification and Editor Routines	119
Verifying Block Count	120
Chapter 5. Unlabeled Tapes	121
Creating an Unlabeled Tape Volume with DFSMSrmm	122
Opening an Input Data Set	122
Positioning the Volume to the Data Set	123
Read Backward	124
End-of-Data or End-of-Volume on Input	124
Determining Volume Switch	124
Opening an Output Data Set	125
Volume Serial Number	126
Positioning the Volume to the Data Set	126
End-of-Volume on Output	127
Closing an Output Data Set	127
Restarting from a Checkpoint	127
Chapter 6. Using Tape Volumes Created by Other Systems	129
IBM Standard Labels	129
Nonstandard Labels	130
Unlabeled Tapes	130
Leading Tapemarks	130
Appendix A. Using the IBM 3495 Automated Tape Library Dataserver to Read Tape Volumes	131
Tape Library	131
Volume Labels	131
Tape Processing	131
Tape Label Verification-Automated Tape Library	131
Specific Mount Request	131
Non-Specific Mount Request	132
Tape Label Creation	132
Appendix B. Using the IBM 3495 M10 Manual Tape Library Dataserver to Read Tape Volumes	135
Tape Volumes	135
Tape Library	135
Volume Labels	135
Message Processing	135
Tape Processing	135
Tape Label Verification-Manual Tape Library	136
Tape Library Scratch Tape Pools	136
Appendix C. Component Support of Label Processing	137
Appendix D. External Labels	139
Reel Label	139
Contents Label	139
Checkpoint Security Label	140
Cartridge Label	140
Appendix E. Equivalent ASCII and EBCDIC Codes	143
ASCII	143

EBCDIC to ASCII	143
ASCII to EBCDIC	146
Conversion Irregularities	147
Abbreviations	149
Glossary	151
Index	153

Figures

1.	Basic Tape Layouts	2
2.	Sources and Sequence for Completing the Data Control Block	4
3.	DEN Parameter Codes for Specifying Tape Density	13
4.	Codes for the TRTCH parameter	14
5.	Codes for the TRTCH parameter	15
6.	Volume Organizations with IBM Standard Labels (Single Data Set) . . .	18
7.	Volume Organizations with IBM Standard Labels (Multiple Data Sets) . .	19
8.	Format of IBM Standard Volume Label	23
9.	Format of IBM Standard Data Set Label 1	26
10.	Format of IBM Standard Data Set Label 2	33
11.	Format of IBM Standard Data Set Label 2 Continue	34
12.	Format of User Label	39
13.	IBM Standard Label Processing by Open/Close/EOV Routines	40
14.	Volume Organization with ISO/ANSI Labels (Single Data Set)	66
15.	Volume Organizations with ISO/ANSI Labels (Multiple Data Sets)	67
16.	Format of the ISO/ANSI Volume Label	72
17.	Format of ISO/ANSI Header 1 and Trailer 1 labels	75
18.	Format of ISO/ANSI Header 2 and Trailer 2 labels	82
19.	Format of ISO/ANSI User Labels	87
20.	Example of a ISO/ANSI Tape from a Non-IBM System and a Corresponding Output Tape	89
21.	ISO/ANSI Label Processing by Open/Close/EOV Routines	91
22.	Organization for Unlabeled Tapes	122
23.	Component Support of Label Processing Types	137
24.	External Label for Reel Identification	139
25.	External Labels for Contents Identification	140
26.	External Label for Cartridge Identification	141

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of the intellectual property rights of IBM may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, programs, or services, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Information Enabling Requests
Dept. DWZ
5600 Cottle Road
San Jose, CA 95193

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Any pointers in this publication to non-IBM Web sites are provided for convenience only, and do not in any manner serve as an endorsement of these Web sites.

Programming Interface Information

This book is intended to help you to use DFSMS/MVS processing of magnetic tape labels. This book primarily documents General-use Programming Interface and Associated Guidance Information provided by DFSMS/MVS.

General-use programming interfaces allow the customer to write programs that obtain the services of DFSMS/MVS.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States, or other countries, or both:

3090	MVS/XA
ADSTAR	NetDoor
AFP	OPC
AIX	OpenEdition
CICS	OS/2
DATABASE 2	OS/390
DB2	PC AT
DFSMS/MVS	Print Services Facility
DFSMSdfp	ProBranch
DFSMSdss	PS/Note
DFSMSHsm	PS/2
DFSMSrmm	PSF
ES/4381	PSF/6000
ESA/370	RACF
ESA/390	Resource Measurement Facility
ESCON	RETAIN
Hardware Configuration Definition	RMF
Hiperbatch	RPG
Hiperspace	RS/6000
IBM	S/370
ImagePlus	System/360
IMS/ESA	System/370
Knowledge Mining Center	System/390
LAN Distance	Systems Application Architecture
MVS/DFP	Ultimotion
MVS/ESA	VSE/ESA
MVS/SP	

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names, denoted by a double asterisk (**), might be trademarks or service marks of others.

About This Book

This book is intended to help you understand and use DFSMS/MVS processing of magnetic tape labels.

This product is designed according to the specifications of the following industry standards as understood and interpreted by IBM as of April 1983:

- *American National Standard Code for Information Interchange*, X3.4-1977. This is a revision of the 1968 version of the code and is given the acronym ASCII.
- *American National Standard Magnetic Tape Labels and File Structure for Information Interchange*, X3.27-1978.
- *American National Standard Recorded Magnetic Tape for Information Interchange, 200cpi, NRZI*, X3.13-1973.
- *American National Standard Recorded Magnetic Tape for Information Interchange, 800cpi, NRZI*, X3.22-1973
- *American National Standard Recorded Magnetic Tape for Information Interchange, 1600cpi, PE*, X3.39-1973
- *American National Standard Recorded Magnetic Tape for Information Interchange, 6250cpi Group-Coding*, X3.54-1976
- *Data Processing—7-Bit Coded Character Set for Information Interchange*, ISO 646-1977
- *Information Processing—Magnetic Tape Labeling and File Structure for Information Interchange*, ISO/DIS 1001-1979
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 8rpm (200rpi)*, ISO 1862
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 32rpm (800rpi)*, ISO 1863
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 8 and 32rpm (200 and 800rpi) NRZI, and 63rpm (1600rpi), Phase Encoded*, ISO 1864
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 63rpm (1600rpi), Phase Encoded*, ISO 3788

Required Product Knowledge

To use this book effectively, you should be familiar with:

- Data management
- Job control language.

Required Publications

You should be familiar with the information presented in the following publications:

Publication Title	Order Number
<i>DFSMS/MVS Version 1 Release 5 Using Data Sets</i>	GC28-1758

Publication Title	Order Number
<i>OS/390 V2R6.0 MVS JCL User's Guide</i>	GC28-1758

How to Tell if this Book is Current

IBM regularly updates its books with new and changed information. When first published, both hardcopy and BookManager softcopy versions of a book are identical, but subsequent updates might be available in softcopy before they are available in hardcopy. Here's how to determine the level of a book:

- Check the book's order number suffix (often referred to as the dash level). A book with a higher dash level is more current than one with a lower dash level. For example, in the publication order number SC26-4930-02, the dash level 02 means that the book is more current than previous levels, such as 01 or 00. Suffix numbers are updated as a product moves from release to release, as well as for hardcopy updates within a given release.
- Check to see if you are using the latest softcopy version. To do this, compare the last two characters of the book's file name (also called the book name). The higher the number, the more recent the book. For example, DGT1U302 is more recent than DGT1U301.
- Compare the dates of the hardcopy and softcopy versions of the books. Even if the hardcopy and softcopy versions of the book have the same dash level, the softcopy could be more current. This will not be apparent from looking at the edition notice. The edition notice number and date remain that of the last hardcopy version. When you are looking at the softcopy product bookshelf, check the date shown to the right of the book title. This will be the date that the softcopy version was created.

Also, an asterisk (*) is added next to the new and changed book titles in the CD-ROM booklet and the README files.

Vertical lines to the left of the text indicate changes or additions to the text and illustrations. For a book that has been updated in softcopy only, the vertical lines indicate changes made since the last printed version.

Related Publications

This book references some publications from the OS/390 library. *OS/390 V2R6.0 Information Roadmap*, GC28-1727, contains a complete list of the OS/390 publications and identifies new publications or publications that replace the ones in the previous version.

Referenced Publications

Within the text, references are made to the following publications:

Publication Title	Order Number
<i>3480 Magnetic Tape Subsystem: User's Reference</i>	GC32-0099
<i>DFSMS/MVS Version 1 Release 5 DFSMSdfp Advanced Services</i>	SC26-4921

Publication Title	Order Number
<i>DFSMS/MVS Version 1 Release 5 DFSMSdfp Storage Administration Reference</i>	SC26-4920
<i>DFSMS/MVS Version 1 Release 5 DFSMSrmm Implementation and Customization Guide</i>	SC26-4932
<i>DFSMS/MVS Version 1 Release 5 General Information</i>	GC26-4900
<i>DFSMS/MVS Version 1 Release 5 Implementing System-Managed Storage</i>	SC26-3123
<i>DFSMS/MVS Version 1 Release 5 Installation Exits</i>	SC26-4908
<i>DFSMS/MVS Version 1 Release 5 Object Access Method Planning, Installation, and Storage Administration Guide for Tape Libraries</i>	SC26-3051
<i>DFSMS/MVS Version 1 Release 5 Using Data Sets</i>	SC26-4922
<i>DFSMS/MVS Version 1 Release 5 Utilities</i>	SC26-4926
<i>OS/390 JES3 Initialization and Tuning Guide</i>	SC28-1802
<i>OS/390 MVS Initialization and Tuning Guide</i>	SC28-1751
<i>OS/390 MVS JCL User's Guide</i>	GC28-1758
<i>OS/390 MVS System Data Set Definition</i>	GC28-1782
<i>OS/390 Security Server (RACF) Introduction</i>	GC28-1912
<i>OS/390 Security Server (RACF) Security Administrator's Guide</i>	SC28-1915

Also see the standards listed in Chapter 3, “ISO/ANSI Labels” on page 63

References to Product Names Used in DFSMS/MVS Publications

DFSMS/MVS publications support DFSMS/MVS, 5695-DF1, as well as the DFSMSdfp base element and the DFSMSshm, DFSMSdss, and DFSMSrmm features of OS/390, 5647-A01. DFSMS/MVS publications also describe how DFSMS/MVS interacts with other IBM products to perform the essential data, storage, program and device management functions of the operating system.

DFSMS/MVS publications typically refer to another IBM product using a generic name for the product. When a particular release level of a product is relevant, the reference includes the complete name of that product. This section explains the naming conventions used in the DFSMS/MVS library for the following products:

MVS can refer to:

- MVS/ESA SP Version 5, 5695-047 or 5695-048
- The MVS base control program (BCP) of OS/390, 5647-A01

All MVS book titles used in DFSMS/MVS publications refer to the OS/390 editions. Users of MVS/ESA SP Version 5 should use the corresponding MVS/ESA book. Refer to *OS/390 Information Roadmap* for titles and order numbers for all the elements and features of OS/390.

For more information about OS/390 elements and features, including their relationship to MVS/ESA SP and related products, please refer to *OS/390 Planning for Installation*.

RACF can refer to:

- Resource Access Control Facility (RACF), Version 2, 5695-039
- The RACF element of the OS/390 Security Server, an optional feature of OS/390

All RACF book titles refer to the Security Server editions. Users of RACF Version 2 should use the corresponding book for their level of the product. Refer to *OS/390 Security Server (RACF) Introduction* for more information about the Security Server.

CICS can refer to:

- CICS/MVS, 5665-403
- CICS/ESA, 5685-083
- The CICS element of the CICS Transaction Server for OS/390, 5665-147

All CICS book titles refer to the CICS Transaction Server for OS/390 editions. Users of CICS/MVS and CICS/ESA should use the corresponding books for those products. Please see *CICS Transaction Server for OS/390: Planning for Installation* for more information.

Summary of Changes

The following sections describe the updates to this book.

Third Edition, March 1999

This edition is a minor revision in support of the functional changes introduced with DFSMS/MVS Version 1 Release 5. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change (a revision bar). For a book that has been updated in softcopy only, the vertical lines indicate changes made since the last printed version.

The following are the changes made to this edition:

- Chapter 2, “IBM Standard Labels” on page 17, the format of IBM standard data set label 2 has been updated. In the Reserved field the number of bytes has been changed to 2 and the Device Serial Number has been changed to 6 bytes.
- Chapter 3, “ISO/ANSI Labels” on page 63, ISO/ANSI labels was updated with information pertaining to ISO/ANSI Version 4 support.
- Chapter 3, “ISO/ANSI Labels” on page 63, ISO/ANSI volume labels VOL2–VOL9 are optional.
- Chapter 3, “ISO/ANSI Labels” on page 63, the format of ISO/ANSI volume label has been updated. The Reserved for Future Standardization field has been changed to 13 bytes and a new field has been added called Implementation Identifier which also has 13 bytes.
- Chapter 3, “ISO/ANSI Labels” on page 63, the format of ISO/ANSI Data Set Label 2 (HDR2/EOV2/EOF2) has been updated. The Reserved for Operating System field in Figure 18 on page 82 has a new field called Coded Character Set Id (CCSID) and it has 5 bytes.

Note: For other important updates to this book, please check informational APAR II11474, a repository of DFSMS/MVS 1.5 information that was not available at the time DFSMS/MVS books were published for general availability.

Chapter 1. Introduction to Tape Processing

Labels are used to identify magnetic tape volumes and the data sets on those volumes. You can process tape volumes with:

- IBM* standard labels
- Labels that follow standards published by:
 - International Organization for Standardization (ISO)
 - American National Standards Institute (ANSI)
- nonstandard labels
- no labels.

Your installation can install a bypass for any type of label processing; however, the use of labels is recommended as a basis for efficient control of your tape volumes.

IBM standard tape labels consist of *volume labels* and groups of *data set labels*. The volume label, identifying the volume and its owner, is the first record on the tape. The data set label, identifying the data set and describing its contents, precede and follow each data set on the volume:

- The data set labels that precede the data set are called *header labels*.
- The data set labels that follow the data set are called *trailer labels*. They are almost identical to the header labels.
- The data set label groups can include standard *user labels* at your option.

Usually, the formats of ISO and ANSI labels, which are defined by the respective organizations, are similar to the formats of IBM standard labels. Unless otherwise specified, the term *standard label*, as used in this manual, refers to IBM, ISO, and ANSI standard labels. However, whereas ISO labeled tapes are coded in the International Standard Code for Information Interchange (ISCII) and ANSI labeled tapes are coded in the equivalent American National Standard Code for Information Interchange (ASCII), IBM labeled tapes are coded either in the extended binary-coded-decimal interchange code (EBCDIC) or in binary-coded-decimal interchange code (BCDIC).

Nonstandard tape labels can have any format and are processed by routines you provide. Unlabeled tapes contain only data sets and tapemarks.

Figure 1 on page 2 shows the IBM standard, ISO and ANSI standard, non-standard, and unlabeled tape layouts for a single data set on a single volume. Detailed layouts and variations for each type are illustrated and described in the appropriate sections of this manual.

Tape volumes with standard tape labels may be defined to RACF* by volume serial under the TAPEVOL class of entities. RACF authorization checking will be performed for every standard labeled tape if system-wide tape protection has been specified. No protection is specifically extended by the system to nonstandard labeled tapes, but the installation-written nonstandard tape label routines may provide such protection.

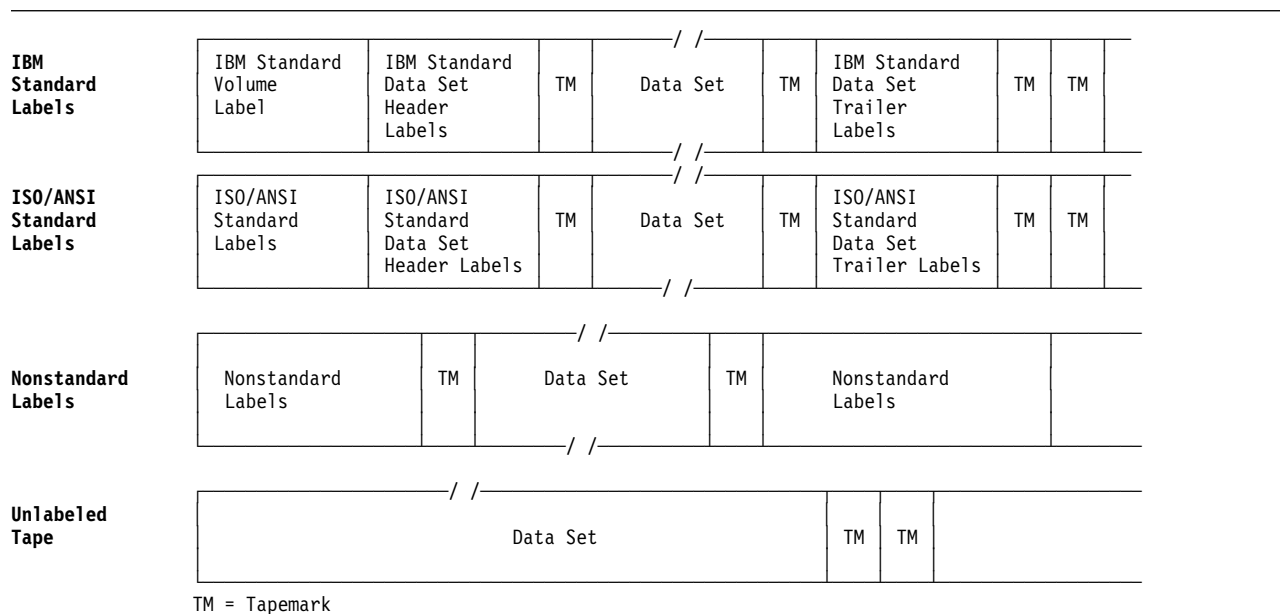


Figure 1. Basic Tape Layouts

Describing the Labels

In the job control statements, you must provide a data definition (DD) statement for each data set to be processed. The LABEL parameter of the DD statement is used to describe the data set's labels. You specify the type of labels by coding one of the following subparameters of the LABEL parameter:

Code	Meaning
SL	IBM standard labels
AL	ISO/ANSI labels
SUL	Both IBM Standard and user header or trailer labels
AUL	Both ISO/ANSI and user header or trailer labels
NSL	Nonstandard labels
NL	No labels
BLP	Bypass label processing
	The data set is treated in the same manner as if NL had been specified, except that the system does not check for an existing volume label. If your installation does not allow BLP, the data set is treated exactly as if NL had been specified. Your job can use BLP only if Job Entry Subsystem (JES), RACF, or DFSMSrmm* allow it.
LTM	Bypass a leading tapemark, if encountered, on unlabeled tapes.

If you do not specify the label type, the operating system assumes that the data set has IBM standard labels.

RACF protects SL, AL, SUL, and AUL tape volumes. The installation may provide support for RACF protection of NSL tapes. When the first data set on the first volume specified in the DD statement is created, the volume may be automatically

defined to RACF if PROTECT=YES is coded in the DD statement. PROTECT=YES may be specified for SL, AL, NL, SUL, AUL, BLP, and NSL tape volumes. For NSL tapes, the first data set on the first volume is not a criterion for a valid PROTECT specification.

The data set sequence subparameter of the LABEL parameter specifies the data set's relative position on the tape. If you do not specify the relative position, the operating system assumes that the data set is first on the reel.

When INOUT, OUTIN, or OUTINX is specified as the processing method in the OPEN macro, the LABEL parameter can override this specification. If INOUT is specified and you want the data set processed for input only, code the subparameter IN in the LABEL parameter. If OUTIN or OUTINX is specified and you want the data set processed for output only, code the subparameter OUT in the LABEL parameter. INOUT is not supported for ISO and ANSI tapes, but is treated as input if the subparameter IN is used in the LABEL parameter.

When new data sets are created, the LABEL parameter is used to record an expiration date and a security protection status in the label. Beginning with RACF 1.7, the LABEL parameter value is also used by RACF to determine the security expiration. If not specified, the expiration date records as zeros, allowing the data set to be overwritten immediately, unless DFSMSrmm is managing the volume. Password protection is not provided.

Describing the Data Sets

Other parameters of the DD statement identify the data set, give volume and unit information and volume disposition, and describe the data set's physical attributes. You can use a data class to specify all of your data set's attributes such as record length and record format but not data set name and disposition. Specify the name of the data class using the JCL keyword DATACLAS. If you do not specify a data class, the automatic class selection (ACS) routines assign a data class based on the defaults defined by your storage administrator. See *DFSMS/MVS DFSMSdfp Storage Administration Reference* for more information on data class.

Allocating Tape Data Sets

An example of allocating a tape data set using DATACLAS in the DD statement follows. TAPE01 is the name of the data class.

```
//NEW DD DSN=DATASET.NAME,UNIT=TAPE,DISP=(,CATLG,DELETE),DATACLAS=TAPE01
```

Completing the Data Control Block

The information contained in the DD statement or a dynamic allocation request is stored in a table called the job file control block (JFCB). Each data set to be processed must also be represented by a data control block (DCB) that is created in storage by the processing program. When completed, the DCB contains full descriptive information about the data set, and is the connection between the data set, the processing program, and the operating system.

The system can determine the optimum block size for new tape data sets. For more information, see "System-Determined Block Size," in *DFSMS/MVS Using Data Sets*.

Most of the information recorded in the DCB is obtained from:

- The DCB macro in the processing program or a DCB created by a compiler. The DCB macro is used to construct a DCB and to provide information about the data set.
- The JFCB, which the system creates from the DD statement in the input stream or from dynamic allocation parameters.
- Data class definition, if any.
- The data set label (if this is an existing data set).

The DCB is completed at execution time, when it is opened. Figure 2 illustrates the sequence of filling in the DCB information. Steps 3 and 8 are bypassed if the tapes have nonstandard labels or no labels.

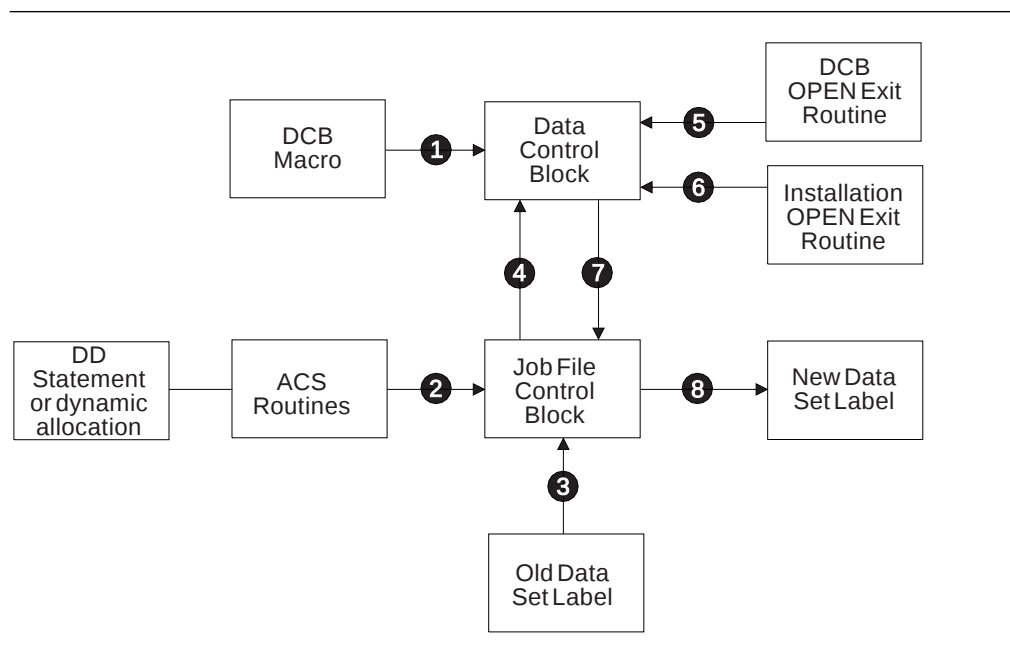


Figure 2. Sources and Sequence for Completing the Data Control Block

Forward Merge (Steps 3 and 4): If the tape data set is being opened for INPUT, INOUT, RDBACK or EXTEND, or it is opened for OUTPUT in conjunction with DISP=MOD, then information from the standard data set label is merged into vacant fields of the JFCB. (Any fields that were already specified by the DD statement are not changed.) Then, for any OPEN option, information from the JFCB is merged into vacant fields of the DCB. (Any fields that were already specified by the DCB macro are not changed.) When the forward merge is completed, your processing program can use the DCB open user exit routine to modify the DCB. For a description of the DCB open user exit routine, see *DFSMS/MVS Using Data Sets*. After the optional DCB OPEN user exit routine, the system calls the installation OPEN exit routine. For more information on installation exits, refer to *DFSMS/MVS Installation Exits*.

Reverse Merge (Steps 7 and 8): After the DCB is completed, the merging process is reversed. For an input data set, information from the DCB is used to fill in any vacant fields of the JFCB. For an output data set, the DCB information overrides

the JFCB information (except the data set organization field), and the updated JFCB provides the information for creating the new labels.

Data Sets Managed by Storage Management Subsystem

If you have the Storage Management Subsystem (SMS) active, you can move tape data sets to DASD to take advantage of the space, performance, and availability capabilities of SMS. If you put these tape data sets on DASD, you will have less tape activity, use fewer tapes, and need less operator intervention. DFHSM can move large numbers of appropriate data sets to tape much more efficiently than if users controlled it directly.

For more information on the Storage Management Subsystem, see *DFSMS/MVS General Information*. Also, *DFSMS/MVS DFSMSdfp Storage Administration Reference* explains how to move tape data sets to DASD.

Data Sets and Volumes Managed by DFSMSrmm

Data sets on magnetic tape and tape volumes can be managed by DFSMSrmm. DFSMSrmm provides validation, overwrite prevention, 'return to scratch', movement control between libraries and locations, and other management functions.

For more information on DFSMSrmm, see *DFSMS/MVS DFSMSrmm Implementation and Customization Guide*.

Cataloged Data Sets

The operating system has facilities that automatically record the following information about each of your data sets:

- The data set name
- The serial numbers of the volume or volumes containing the data set
- The type of device on which the volumes should be mounted
- The data set's relative position on its first volume.

The information is indexed by the data set name and recorded on a direct access device in a logical structure called the catalog. You can retrieve a cataloged data set by specifying its name in the DD statement and omitting the volume serial. The system finds the associated information in the catalog and issues a command, to either the operator or the Tape Library Dataserver, to mount the cartridge on a drive.

Generation Data Groups

A cataloged data set that is frequently updated, such as a weekly payroll, can be grouped with its earlier generations to form a named generation data group. A lower-level index in the catalog structure allows generation and version numbers to be included in the data set name. For example, the original generation of the data set group A.PAYROLL is named A.PAYROLL.G0001V00. The fourth generation of the data set is identified as A.PAYROLL.G0004V00. The absolute generation and version numbers are in the form GxxxxVyy, where:

- xxxx is a decimal number (0001 to 9999) showing the relationship to the original generation. The maximum number of generations that can be cataloged is established when the index is built for the particular generation data group.
- yy is a decimal number (00 to 99) identifying a version of the same generation. Only the latest version is cataloged.

You usually refer to a generation data set by specifying its relative generation number. For example, A.PAYROLL(0) refers to the latest cataloged generation; A.PAYROLL(-1) refers to the next-to-the-latest generation; and A.PAYROLL(+1) refers to a new generation to be added to the group.

There are several ways to specify attributes for a new generation data set:

- When a generation data group base is established, the GDG is cataloged. The GDG can be cataloged on either an SMS or non-SMS managed volume. If the catalog resides on an SMS managed volume, then the model DSCB can not exist. (All SMS data sets must be cataloged and the model DSCB has the same name as the GDG. Therefore the model DSCB can not be defined on an SMS volume because its name would conflict with the existing GDG name.) If the catalog resides on a non-SMS managed volume, then the model DSCB can exist; however, if the generation is SMS-managed, then it will not use the model DSCB. The LIKE, REFDD, and DATACLAS attributes can be used for both SMS and non-SMS generation data sets.
- Only explicitly specified DCB attributes, REFDD, LIKE, and DATACLAS are used with SMS to allocate a new generation data set. If a model DSCB exists, it is ignored. However, model DSCBs are still valid. The LIKE keyword specifies the allocation attributes of a new data set by copying the attributes of a cataloged model data set. The following attributes are copied from the model data set to the new data set:
 - Record format (RECFM)
 - Record length (LRECL)
 - Data set organization (DSORG)
 - Record organization (RECORG) (to VSAM only)
 - Key length (KEYLEN) (to DASD only)
 - Key offset (KEYOFF) (to VSAM only)
 - Type, PDS, or PDSE (DSNTYPE) (to DASD only)
 - Space allocation (AVGREC and SPACE) (to DASD only)

Examples of allocating a tape generation data set by supplying its DCB attributes through DATACLAS and LIKE are:

```
//DDNAME DD DSN=HLQ.----.LLQ(+1),DISP=(NEW,CATLG),DATACLAS=dc_name
```

```
//DDNAME DD DSN=HLQ.----.LLQ(+1),DISP=(NEW,CATLG),LIKE=dsname
```

If you do not specify a block size in the JCL, the system calculates a block size. Note that the block size is not obtained from the data class or LIKE data set. The expiration date from the data class is stored in the JFCB. Note that the expiration date from the LIKE data set is not used. For further information, see “System-Determined Block Sizes” in *DFSMS/MVS Using Data Sets*.

Information on creating and retrieving generation data groups can be found in *DFSMS/MVS Installation Exits*, *OS/390 MVS JCL User's Guide*, and *DFSMS/MVS Utilities*.

Concatenated Data Sets

Through the technique of concatenation, several data sets, each of which may reside on a separate volume, can be read as if they were a single data set.

Concatenated data sets are read in the order of appearance of their DD statements in the input stream (the DD statements must follow one another and only the first DD statement is named). Each concatenated data set may be a single- or multi-volume data set. Concatenated data sets cannot be read backward.

Because only one DCB is associated with all the concatenated data sets, you must inform the system if the data sets have certain allocation characteristics that do not match (such as record format). To do this, your processing program must set the unlike-attributes bit in the DCB, as explained in *DFSMS/MVS Using Data Sets*. If the EXCP, POINT, or CNTRL macros are not used, you can concatenate tape, DASD, and sysin data sets if they have certain allocation characteristics that match. In this case, you do not have to set on the unlike-attributes bit in the DCB.

When standard or ANSI labels, buffers gotten by OPEN, and the queued sequential access method (QSAM) are used together, the data set unlike characteristic of “block length” does not apply, and you can concatenate data sets in any order of block size. Normal DCB, JFCB, and label merging are performed to obtain the tape block size when labels are not present or are bypassed.

More information on concatenated data sets can be found in *DFSMS/MVS Using Data Sets*.

Passed Data Sets

When a data set is used by two or more job steps in the same job, you can pass the data set from job step to job step. In this way, you can conveniently refer to the data set in the DD statements for each of the later steps, which are called *receiving steps*. Device type, volume serial numbers, data set sequence number, and label type need not be coded in the DD statements for the receiving steps, because this information is obtained from the passing step. However, the data set attributes (density, record format, and so forth) are not automatically passed to the DD statements in the receiving steps. If the data set has standard labels, the receiving steps can obtain the attributes from the labels. If the data set does not have standard labels, and the processing program does not define the data set attributes, then the DD statements in the receiving steps should restate the attributes. If you use a volume reference, all volumes must be in the same library or the request fails. The 3495 Tape Library Dataserver ensures that unit affinity requests are assigned to the same library of devices.

Multiple Volumes and Multiple Data Sets

To place a single data set on multiple volumes, code the multiple volume serial numbers in the related DD statement, or request a *nonspecific volume*. If you request *specific volumes* and cataloging, all the specified volume serial numbers will be associated with the new data set in the catalog. If you use fewer volumes than you specify, you can not retrieve the data set properly through use of the catalog.

To place multiple data sets on a single volume, code the same volume serial number on each of the related DD statements, or use the VOLUME=REF parameter on the DD statements for the second and subsequent data sets volume.

When you create or retrieve a data set of file sequence greater than one, you must use the LABEL parameter to specify the data set's sequence number. This is not required if retrieval is accomplished through the catalog.

To place multiple data sets on multiple volumes, code a set of volume serial numbers on each of the related DD statements, or use the VOLUME=REF parameter.

Note: VOLUME=REF can not be used if you specified volume serial numbers for the referred-to data set and that data set was written on less volumes.

If you code a set of volume serial numbers for each of the data sets, the system works most efficiently if the first serial number is for the last volume occupied by the preceding data set. This is necessary only if the volumes do not contain standard labels.

For multiple data sets on multiple volumes, you must use the LABEL parameter to specify the sequence number of each data set, both when you create it and when you retrieve it, except when retrieval is accomplished through the catalog. The sequence number specified for each data set must indicate the position of the data set relative to the entire multi-volume group. You can retrieve data sets in an order that differs from the order in which they were written. For SL and AL, OPEN reads the existing labels to determine their position within the volume group. From existing labels, OPEN calculates the appropriate position for the data set being opened. For unlabeled tapes, the system can not determine data set positions in an existing multi-volume group after the first volume. Therefore, you should not use the catalog to retrieve such data sets.

All data sets on a RACF-protected volume are protected. Data sets that span multiple volumes should have all volumes protected under a single RACF tape volume set profile. If a data set on a RACF-protected volume is extended to a new volume that is not RACF-protected, the new volume is defined to RACF as part of the same volume set as the previous volume. Therefore, RACF protection is automatically extended to the new volume.

Processing Methods and Routines

You can specify the method of processing (INPUT, OUTPUT, EXTEND, RDBACK, INOUT, OUTIN, or OUTINX) by using an operand of the OPEN macro. If you do not specify the method, INPUT is assumed.

A data set can be processed as either input or output (INPUT, OUTPUT, or EXTEND). A data set on magnetic tape can also be read backward (RDBACK). If the basic sequential access method (BSAM) is used, a data set can also be processed as a combination of input and output (INOUT or OUTIN, or OUTINX). For INOUT, the data set is an input data set first and then, without reopening, an output data set. For OUTIN or OUTINX, the data set is an output data set first and then, without reopening, an input data set. INOUT is not supported for ISO and ANSI tapes.

Open/Close/End-of-Volume Routines

The open/close/end-of-volume routines of the system perform the label processing.

Opening a Data Set

The open routine is entered when the processing program issues an OPEN that is generated by a macro or by a compiler. The open routine completes the specified DCB, and prepares and positions the data set for processing. It analyzes input header labels (or trailer labels if the tape is read backward), or creates output header labels. If DFSMSrmm is in use, additional tape volume validation is performed and information recorded in the DFSMSrmm control data set.

Closing a Data Set

The close routine is entered when the processing program issues a CLOSE macro. If the processing program terminates without closing the data set, the operating system calls the close routine, which restores the fields of the DCB to the conditions that existed before the data set was opened. It also logically disconnects the data set from the processing program, and creates output trailer labels, and provides for tape disposition. If DFSMSrmm is in use, statistics about processing are recorded in the DFSMSrmm control data set.

End of Data Set or Volume

The EOVS routine is entered when a tapemark is read, when the end of volume is encountered while writing, or when the processing program issues a force-end-of-volume (FEOV) macro. If you use the execute channel program (EXCP) technique, your processing program must issue an EOVS macro to give control to the EOVS routine after your program recognizes a tapemark or end of reel. The EOVS routine processes trailer labels on the current volume (or header labels if the tape is read backward), and determines if additional volumes are needed to continue the data set. If another volume is needed, or another data set is concatenated, then the EOVS routine handles the volume switching and processes the labels on the new volume. Otherwise, if the current volume is the last or only volume needed, then EOVS gives control to the user's end-of-data routine that is specified in the DCB.

If DFSMSrmm is in use, then the function performed at OPEN and CLOSE is performed at EOVS as well.

Checkpoint/Restart Routine

When a job step is restarted from a checkpoint, the restart routine repositions tape volumes containing data sets that were open at the time the checkpoint was taken. The restart routine also restores the applicable DCBs to the conditions that existed when the checkpoint was taken.

The restart routine can handle tapes with IBM standard labels, nonstandard labels, or no labels.

Note: Tapes with ISO/ANSI labels are not supported by checkpoint/restart as checkpoint volumes.

All VSE/ESA* tapes having either a leading tapemark or embedded checkpoint records can be handled by checkpoint/restart, except for VSE/ESA 7-track tapes written in translate mode that contain embedded checkpoint records.

Automatic Volume Recognition

Under automatic volume recognition (AVR), the operator can premount volumes on any unused drives. The volumes must be labeled (standard or nonstandard). The system records the volume and unit information, and assigns the drives when that volume is requested.

Note: If the drive is needed earlier for another request, then the volume will be unloaded.

AVR checks the tape label during allocation by the scheduler, and records the volume serial number. This action merely determines which volumes are mounted on which devices; AVR does not verify or reject the volumes based on their serial numbers. The following conditions will cause the volume to be unloaded:

- volume is unlabeled
- volume has a nonstandard label and either there was no installation exit or the installation exit rejected the volume
- volume serial number is a duplicate of one that is already mounted

In an automated library, a mismatch between the external volume label and the tape volume label causes an error. When AVR detects an error, it causes the volume to be unloaded with a message identifying the error.

Tape Disposition

Tape disposition at the end of the data set or volume can be affected by the DISP parameter of the DD statement. This implied disposition can be overridden by a positioning parameter of the OPEN, FEOV, or CLOSE macro. The OPEN macro controls positioning after an end-of-volume condition (multi-volume data sets), unless overridden by the FEOV macro. The CLOSE macro controls positioning at the end of the data set.

The positioning parameters of the OPEN, FEOV, and CLOSE macro instructions are:

LEAVE	Position the volume at the logical end of the data set just read or written. If the data set has been read backward, the logical end is the physical beginning of the data set.
REREAD	Position the volume at the logical beginning of the data set just read or written. When the data set exists on more volumes than there are units available, the REREAD parameter should not be used with the OPEN macro instruction—it may adversely affect the time required to mount the tapes. REREAD cannot be specified for FEOV.
REWIND	Rewind the volume to the load point. REWIND cannot be specified for OPEN or CLOSE TYPE=T.
DISP	Perform the disposition processing that was requested. This can be REWIND, or REWIND and UNLOAD, depending on the volume attributes. DISP cannot be specified for FEOV.

The CLOSE macro has an additional operand that allows you to release tape data sets and the volumes on which they reside:

FREE Release the data set associated with this DCB. FREE specifies that the data set is to be released for use by another job and that the device on which the data set is mounted can be freed for allocation to another job.

DISP is the default.

If necessary, the specified volume disposition can be overridden by the system. However, you need not be concerned; the system automatically requests the mounting and demounting of volumes depending on the availability of devices at a particular time.

Verifying Block Count

When a program finishes reading a data set that is on a tape (reel or cartridge), the system checks that the expected number of blocks was read. The system does this if the volume contains IBM standard or ISO/ANSI labels and the program uses BSAM, QSAM, or EXCP. With EXCP, the program maintains the block count in the DCB and the programmer codes REPOS=Y on the DCB macro. For more information, see “Verifying Block Count in Label When Reading” on page 50, and “Verifying Block Count in Label When Reading” on page 109.

When a program finishes reading or writing data on a cartridge (not a reel), the system does additional block count checking unless the program is using EXCP and the programmer did not code REPOS=Y on the DCB macro. The system can check for this on all cartridges, including unlabeled tapes. This type of block count checking is independent of the checking that is done only on labeled tapes.

The additional checking is to compare the count accumulated in the data control block with a count that the system calculates based on information maintained by the tape subsystem (hardware). If the two counts differ, the system does not call the block-count-unequal exit. Instead, the system calls the optional DCB ABEND exit for abnormal completion code 117 or 237. If the exit is not defined, or if it does not ignore or recover, then the system issues an ABEND macro. For more information on DCB ABEND exit, see *DFSMS/MVS Using Data Sets*. For more information on the installation DCB ABEND exit, see *DFSMS/MVS Installation Exits*.

For more information on verifying block count when using EXCP, refer to *DFSMS/MVS DFSMSdfp Advanced Services*.

Bypassing Label Processing

If you do not want the system to verify the label type, specify BLP (instead of NL) in the LABEL Parameter of the DD statement. A loadpoint read will be done to check for existing labels. If labels are present, the system will still accept the tape as if NL were specified.

There are other reasons for using the BLP option. For example, you may want to overwrite a 7-track tape that differs from your current parity or density specifications. If such a tape is mounted, the system makes four attempts to read the initial record (to determine that it is not an IBM standard label) before accepting the tape. Each read may result in a long error recovery attempt. You can eliminate the four read operations by specifying BLP instead of NL.

There are some situations where the system will detect that a particular tape may not be the correct volume. Your job may have requested a specific volume or a non-specific volume. If your job requests that label processing be bypassed in these cases, the system demounts the existing tape and asks for a specific or non-specific volume. This action gives the tape operator the chance to insure that the requested volume is mounted.

When a job requests that label processing be bypassed, the OPEN or EOVR routines will demount the currently mounted tape if the drive is in a ready state and any of the following is true:

- The tape has been processed by automatic volume recognition (AVR).
- The tape has been processed by a MOUNT command.
- The mounted volume has been passed or retained from a prior step of the same job.
- The mounted volume is still mounted after having been processed in the same step.

Controlling the Use of BLP

If an installation plans to use RACF protection for tape volumes or password protection for tape data sets, the BLP JCL option must either be disallowed completely, or restricted to authorized users. The BLP option is associated with job class; that is, the installation has the option to allow or disallow BLP by job class. It is recommended that BLP be disallowed completely, or if this is not possible, that one job class be set aside as the “BLP” and controlling class. This can be done by using JES initialization options to allow BLP for only that controlling class and by using installation JCL exits to force TYPERUN=HOLD for all jobs specifying that class. The system operator can then monitor that class and release only jobs authorized to use BLP. For the method of specifying BLP for the JES reader, see *OS/390 JES2 Initialization and Tuning Reference* and *OS/390 JES3 Initialization and Tuning Reference*.

A system programmer can specify that RACF is to control which users can use the BLP option. This is done by activating the TAPEVOL class and defining profile ICHBLP in the FACILITY class. For further information see *OS/390 Security Server (RACF) Security Administrator's Guide*.

For more information on additional controls over some BLP usage, see *DFSMS/MVS DFSMSrmm Implementation and Customization Guide*.

Tape Characteristics

The following paragraphs describe the data recording characteristics of magnetic tape. The discussion includes density, parity, number of tracks, translation, conversion, and tapemarks. The error conditions that can result from conflicting tape characteristics are explained under “IBM Standard Volume Label Verification and Editor Routines” on page 43, “ISO/ANSI Volume Label Verification and Editor Routines” on page 94, and “Nonstandard Volume Label Verification and Editor Routines” on page 119.

MVS* supports the IBM 3420 Magnetic Tape Unit Models 3, 4, 5, 6, 7, and 8, the IBM 3422 Magnetic Tape Subsystem, the IBM 3424 Magnetic Tape Unit (available

only in Brazil, S.A.), the IBM 3430 Magnetic Tape Subsystem, the IBM 3480 Magnetic Tape Subsystems, and the IBM 3490 Magnetic Tape Subsystem.

Except where stated otherwise in this book, labels written on all types of drives are in the same format.

Nine-Track Tapes and Dual Density Feature

The operating system supports 9-track tape reels in densities of 800 bits per inch (bpi), 1600 bpi, and 6250 bpi.

The dual-density feature, available on some tape units, permits the unit to read and write in either of two densities. The density combinations available are 800/1600 bpi and 1600/6250 bpi. You specify the density with the DEN parameter of the DD statement or DCB macro instruction (Figure 3 shows the DEN parameter codes). If the DEN parameter code is not specified, the greater density is assumed.

Figure 3. DEN Parameter Codes for Specifying Tape Density

DEN Value ¹	Recording Density	
	7-track	9-track
1	556	—
2	800	800 (NRZI) ²
3	—	1600 (PE) ³
4	—	6250 (GCR) ⁴

Notes:

1. If the DEN parameter is not supplied by any source, the highest applicable density is assumed.
2. Nonreturn-to-zero inverse mode
3. Phase encoded mode
4. Group coded recording mode

The DEN parameter is ignored for input. The system automatically sets the tape unit to the density of the tape.

For output with dual density, the tape is written in the density you specify or, if unspecified, the default density. If your request is for a standard labeled tape, and the label of the mounted volume is written in the wrong density, the system will rewrite the label to agree with your specification, if you are opening the first data set on the volume. If this is not the first data set on the volume, the system will change your density specification to agree with the density of the volume. Tapes created with the dual-density feature and those created without it are interchangeable.

Seven-Track Tapes

The 7-track feature allows a tape unit to read and write 7-track tapes on reels. This special feature consists of a 7-track read/write head (instead of a 9-track head) and control unit changes, including a translator. Data can be read or written in densities of 556 or 800 bpi with either odd or even parity. Nine-track tapes cannot be read on tape units with the 7-track feature installed. The translator takes 8-bit EBCDIC characters from your buffer and writes them as 6-bit binary coded decimal inter-

change code (BCDIC) tape characters and translates the opposite way during a read operation. Density and parity can be set and the translator can be turned on and off, by mode setting control commands. When the translator is off, only the 6 low-order bits of the characters in your buffer are written on tape; during reading, the 2 high-order bits are set to zeros.

ISO and ANSI standards do not include a specification of 7-track magnetic tape for information interchange. Therefore, the 7-track feature is not applicable for tapes with ISO or ANSI labels.

The data conversion feature can also be installed with the 7-track feature. The data conversion feature makes it possible to write binary data on 7-track tape. It writes 3 characters from your buffer as 4 tape characters, and converts the opposite way during reading. Conversion is turned on and off by mode setting control commands and is mutually exclusive with translation. You must use the data conversion feature to process format-V (variable-length) tape records because the length field of such records contains binary data. You cannot use the data conversion feature with the read backward (RDBACK) processing method.

The operating system supports the various densities of the 7-track feature. You specify the density with the DEN parameter of the DD statement or DCB macro (Figure 3 on page 13 shows the DEN parameter codes). If not specified, the default value is 800 bpi.

The job step will be abnormally terminated if the DEN parameter specifies a density incompatible with the tape unit.

Mode information other than density is specified with the tape recording technique (TRTCH) parameter of the DD statement or DCB macro instruction. The codes for the TRTCH parameters are listed in Figure 4.

Figure 4. Codes for the TRTCH parameter

Code	Meaning
T	Odd parity with translation
C	Odd parity with conversion
E	Even parity with no translation or conversion
ET	Even parity with translation
null (entire parameter is omitted)	Odd parity with no translation or conversion (same as 9-track)

You use the DEN and TRTCH parameters (or their default values) to specify the density and mode of the data to be read or written. If the tape contains standard labels, the DEN parameter also specifies the density of the labels. IBM recommends that all data sets on a tape containing standard labels be written in the same density. IBM standard labels on 7-track tapes are always written in BCD, with the translate bit on, and even parity, regardless of the value of the TRTCH parameter.

Nonstandard labels on 7-track tapes can be read or written in any code with any parity. The density of the labels need not be the same as the density of the data, but the density of associated tapemarks should be carefully planned. System

recognition of tapemarks is ensured only when they are read in the density in which they were written.

Tape Cartridges

The operating system supports several IBM magnetic tape subsystems, such as the IBM 3480 and 3490 Magnetic Tape Subsystems, which use tape cartridges instead of tape reels. For these subsystems, only a single density is available for reading and writing; any density specified with the DEN parameter is ignored.

A magnetic tape subsystem that uses tape cartridges and has Improved Data Recording Capability can record data in either of two formats: compacted format or standard, uncompact format. The recording format is specified in the TRTCH parameter of the JCL DD statement, the dynamic allocation text units, or the DCB macro. Figure 5, lists the parameters.

Figure 5. Codes for the TRTCH parameter

Code	Meaning
COMP	Record data in compacted format
NOCOMP	Record data in standard, uncompact format
null (entire parameter is omitted)	Record data in default format

Tapemarks

A data set or label group on tape is usually followed by a tapemark delimiter. A tapemark is a special character written by a control command. (In the figures of this manual, tapemark is represented as TM.) The tape drive recognizes a tapemark during a read operation and signals a unit exception condition to the operating system. The tapemark is not read into virtual storage.

Beginning and End of Tape

On 7-track and 9-track tape units, a reflective strip at the beginning of a tape indicates when the tape is positioned at its *load point*; a load point indicator bit is set in a sense byte, and is recognized by the operating system. A reflective strip also marks the logical end of the tape. If a reflective strip is encountered during a write operation, the hardware signals a unit exception condition in the interrupt to the processor.

Some IBM magnetic tape subsystems, such as the IBM 3490, use tape cartridges instead of tape reels. Subsystems that use tape cartridges have internal mechanisms that sense the beginning and end of tape, and do not use reflective strips.

Chapter 2. IBM Standard Labels

If you specify SL in the LABEL parameter of the DD statement, or if you do not specify a label type, the system performs IBM standard label processing. If you specify SUL, the system processes both IBM standard labels and IBM standard user labels.

This chapter describes the organization, formats, and contents of IBM standard labels and IBM standard user labels. It also explains how the labels are processed or created, and the use of these labels in processing data sets.

Label Definitions and Organization

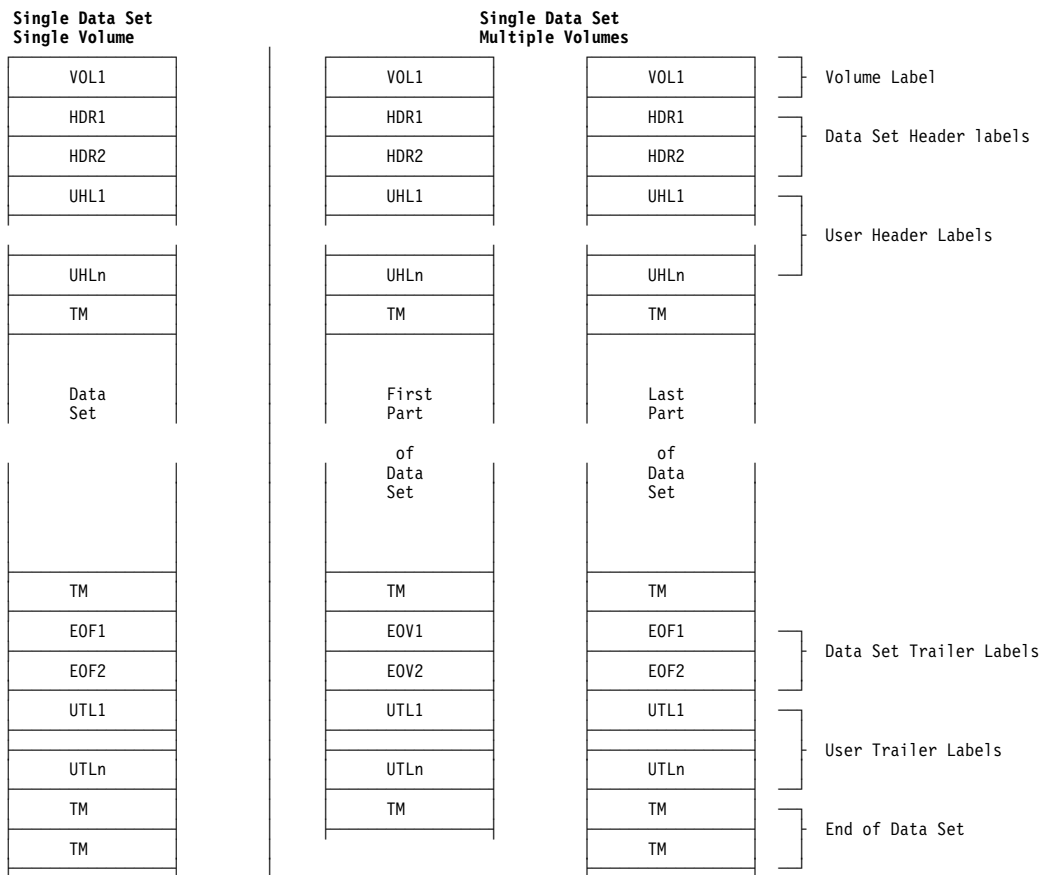
IBM standard labels are 80-character records written in the density specified via JCL. Cartridge and nine-track tape label records are recorded in EBCDIC. Seven-track tape label records are recorded in BCDIC, even parity, translate on. The first 4 characters are always used to identify the labels:

Label Identifier	Label Description
VOL1	Volume label
HDR1 and HDR2	Data set header labels
UHL1 through UHL8	User header labels
EOV1 and EOV2	Data set trailer labels (end-of-volume)
EOF1 and EOF2	Data set trailer labels (end-of-data-set)
UTL1 through UTL8	User trailer labels

The header and trailer labels use identical formats; therefore, there are only four different label formats. These formats are described later in this section. The four types are:

- Standard Volume Label (identified as VOL1)
- Standard Data Set Label 1 (identified as HDR1, EOV1, or EOF1)
- Standard Data Set Label 2 (identified as HDR2, EOV2, or EOF2)
- Standard User Label (identified as UHL1-UHL8 or UTL1-UTL8).

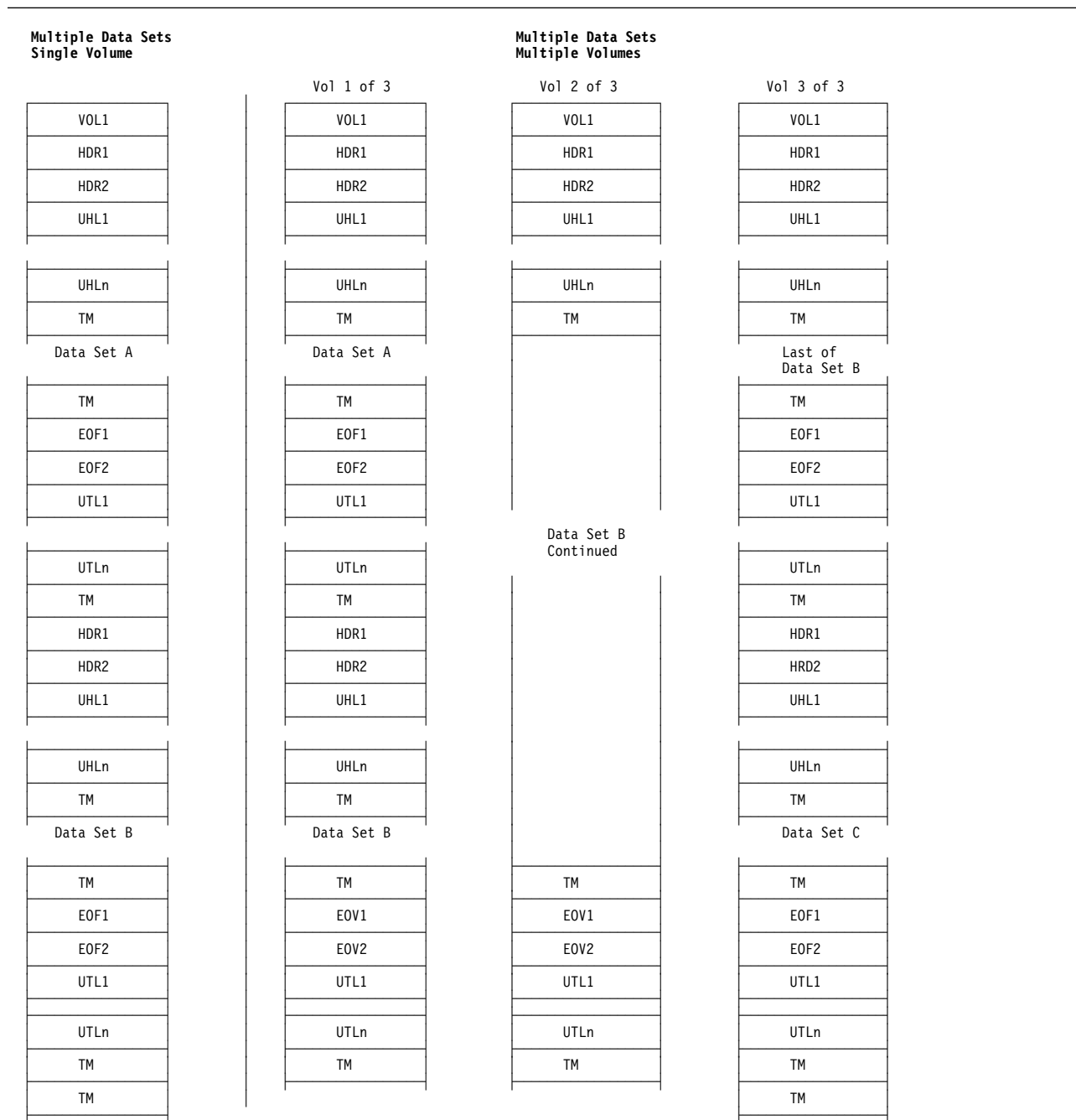
Figure 6 on page 18 and Figure 7 on page 19 show the positions of the labels with various tape volume organizations. A tape with IBM standard labels must contain a volume label and data set labels. User labels are optional.



Single Data Set/Single Volume: The volume label is followed by the data set header labels and optional user header labels. The data set is preceded and followed by a tapemark. The data set trailer labels are identified as EOF and followed by optional user trailer labels. Two tapemarks follow the trailer label group to indicate that the data set is the last data set on the volume and is not continued on another volume.

Single Data Set/Multiple Volumes: More than one volume is needed to contain the data set. The last volume is organized the same as a single volume. On the other volumes, the data set trailer labels are identified as EOV instead of EOF, and the trailer label group is followed by one tapemark instead of two. The data set and user labels are repeated on each volume, and there is a separate volume label for each tape.

Figure 6. Volume Organizations with IBM Standard Labels (Single Data Set)



Multiple Data Sets/Single Volume: The tape begins with a volume label. Each data set is preceded by a header label group and a tapemark, and is followed by a tapemark and a trailer label group. The data set trailer labels are identified as EOF. Each trailer label group is followed by a tapemark; the trailer label group for the last data set on the volume is followed by two tapemarks.

Multiple Data Sets/Multiple Volumes: More than one volume is needed to contain the multiple data set aggregate. The last volume is organized the same as a multiple data set/single volume layout. On the other volumes, the last data set trailer labels are identified as EOV instead of EOF, and the last trailer label group is followed by one tapemark instead of two. There is a separate volume label for each tape.

Figure 7. Volume Organizations with IBM Standard Labels (Multiple Data Sets)

Volume Label

The IBM standard volume label (VOL1) appears at the beginning of each tape. The volume label identifies the volume and its owner and is used to verify that the correct volume is mounted.

Volume labels are created by the EDGINERS or IEHINITT utility programs, or by input from the operator during the Open or EOVS routine when:

- A label conflict is detected (a label conflict occurs when the type of label requested differs from the type of label read at load point from the mounted volume)
- The first block of a volume cannot be read (except a RACF-protected volume is rejected if failure to read was because the tape format, such as a nonsupported density, was not recognizable by the control unit).

For additional information about EDGINERS and the IEHINITT utility programs, see *DFSMS/MVS DFSMSrmm Implementation and Customization Guide*.

Data Set Header Labels

The data set header label group consists of IBM standard data set label 1 (HDR1) and IBM standard data set label 2 (HDR2). The HDR1 label contains operating system and device-dependent data that relates to the data set. The HDR2 label is optional on input and contains additional data set characteristics. These labels are used to identify and describe the data set and to protect it from unauthorized use.

These labels are created automatically by the system each time a data set is recorded on tape.

User Header Labels

Optionally, a maximum of eight user header labels (UHL1 to UHL8) can appear on the tape immediately following the data set header labels. These labels contain user-specified data that can be made available to your program for processing.

If you want the system to write user header labels or to make user header labels available to your program, you must specify SUL on the DD statement and specify the address of a user header label routine in the DCB exit list. (The exit list can address several user header label routines, that is, routines that process input user header labels and create output user header labels.) The DCB exit list is described in *DFSMS/MVS Using Data Sets*.

Data Set Trailer Labels

The data set trailer label group consists of IBM standard data set label 1 (EOV1 or EOF1) and IBM standard data set label 2 (EOV2 or EOF2). These labels duplicate the IBM data set header labels so that the tape can be read backward. The trailer labels are identical to the header labels, except that:

- The identifier is EOV or EOF instead of HDR.
- A block count is recorded in the first trailer label (EOV1 or EOF1) and is used on input to verify that all blocks of the data set are processed. The block count field in the HDR1 label contains zeros (EBCDIC or BCDIC).

These labels are created automatically by the system when the data set is recorded on tape.

User Trailer Labels

Optionally, a maximum of eight user trailer labels (UTL1-UTL8) can immediately follow the data set trailer labels. These labels contain user-specified data that can be made available to your program for processing.

If you want the system to write user trailer labels or to make user trailer labels available to your program, you must specify SUL on the LABEL parameter of the DD statement and specify the address of a user trailer label routine in the DCB exit list. The exit list can address several user trailer label routines, that is, routines that process input user trailer labels and create output user trailer labels. The DCB exit list (EXLST) is described in *DFSMS/MVS Using Data Sets*.

Additional Labels

The operating system does not support any additional labels in the groups described above. This applies to labels identified as VOL2-VOLn, HDR3-HDRn, UHL9-UHLn, and so forth. If such labels exist on an input tape, they are bypassed. They are omitted on output tapes.

System-Managed Mountable Tape External Labels

An external label is required for every tape cartridge in the tape library. This label contains both machine-readable and operator-readable identification characters that are used by the Tape Library Dataserver vision system. A Tape Library Dataserver vision system, which is unique to an automated tape library, identifies the volume serial number of the cartridge. In a manual tape library, a machine readable external label is not required, but is advisable for migration purposes. Each label can have a maximum of six uppercase characters for identification purposes:

- Uppercase A-Z
- Numerics 0-9
- Trailing blank space (non-service volumes only)

The external label must be applied within the indentation on the side of the cartridge; however, a larger information label, supplied by the customer, can be used on the top of the cartridge.

Tapemarks

Each data set and each data set label group to be processed by data management must be followed by a tapemark.

- There is no tapemark between the volume label and the first header label group on the volume.
- The tapemark that marks the end of the header label group also indicates the beginning of the data set to be processed.
- The tapemark that follows the data set also indicates the beginning of the trailer label group.
- A tapemark marks the end of the trailer label group. A second tapemark follows the trailer label group of the last data set on the volume, provided that the data set does not continue on another volume.

When the operating system is used to create a data set with IBM standard labels, the system writes the necessary tapemarks.

IBM Standard Labels Format

Format of the IBM Standard Volume Label (VOL1)

The IBM standard volume label (VOL1) is 80 characters in length and is used to identify the tape volume and its owner. It is always the first record on an IBM standard labeled tape. It is recorded in EBCDIC on 9-track tape units, or in BCDIC on 7-track tape units.

Figure 8 on page 23 shows the format of the volume label. The shaded areas represent fields that are recorded in the label, but are not used or verified during processing. The contents and processing of each field of the label are described below. The processing descriptions refer to the following system control blocks:

- Job file control block (JFCB)
- Unit control block (UCB).

Position	(Bytes)	Field Number and Name
1	(3)	1 Label Identifier ("VOL")
3		
4	(1)	2 Label Number ("1")
5	(6)	3 Volume Serial Number
10		
11	(1)	4 Reserved
12	(5)	5 VTOC Pointer (Direct Access Only)
16	(25)	6 Reserved
17		
41		
42	(10)	7 Owner Name and Address Code
51	(29)	8 Reserved
52		
80		

☐ Functional field.

☐ No Processing function at the present time.

Figure 8. Format of IBM Standard Volume Label

1—Label Identifier (3 bytes)

- *Contents:* The characters VOL identify this label as a volume label.

- *Processing:* This field is read to verify that a standard labeled tape is mounted, and that this label is a volume label.

The labels are created initially by the DFSMSrmm-supplied EDGINERS utility program, the IBM-supplied IEHINITT utility program. IEHINITT can not create labels on tape volumes in a Tape Library Dataserver. Although IEHINITT is not supported in an automated tape library dataserver, it is supported in a manual tape library dataserver.

The labels are created initially by the IEHINITT utility program.

For standard labeled tape volume requests, if OPEN or EOVS enters label editor routines, the VOL1 label can be written or rewritten. See “IBM Standard Volume Label Verification and Editor Routines” on page 43 for more information.

2—Label Number (1 byte)

- *Contents:* The relative position of this label within a set of labels of the same type; it is always 1 for the IBM standard volume label.
- *Processing:* Verified in conjunction with Field 1 to identify this label as VOL1.

3—Volume Serial Number (6 bytes)

- *Contents:* A unique identification code that is assigned through DFSMSrmm or IEHINITT to the volume when it enters the system, or that is assigned by the operator when the Open or EOVS routines label the volume. This code may also appear on the external surface of the volume for visual identification. The code is normally numeric characters (000000 to 999999), but may be any 6 alphanumeric characters. It may be from 1 to 6 characters, but, if fewer than 6 characters, the code must be left-justified, and the remainder is padded with blanks.

If the volume serial number is specified in the JCL statements, all national characters, the hyphen, and other special characters are accepted when enclosed in apostrophes. Their use is not recommended, however, because difficulties can occur in recognizing volume serial numbers when typewriter heads and print chains with nonalphanumeric characters are used.

If the volume serial number is assigned through the IEHINITT utility program, A through Z, 0 through 9, and the hyphen are the only valid characters that may be specified.

In a Tape Library Dataserver there is an additional restriction. See “System-Managed Mountable Tape External Labels” on page 21.

- *Processing:* When the user specifies a volume serial number, it is in the JFCB. The system copies it to the UCB. When the system processes the labels, it compares the number in the UCB to the number in this field of the label to ensure that the correct volume is mounted.

For scratch output tapes, the volume serial number is obtained from this field of the label and recorded in both the JFCB and the UCB.

The EDGINERS and IEHINITT utility programs can create this label with a volume serial number of up to 6 characters. The number is left-justified, and the remainder of this field is padded with blanks.

4—Reserved (1 byte)

- *Contents*: Reserved for possible future use—initialized as a blank or zero.
- *Processing*: Not used.

5—VTOC Pointer (5 bytes)

- *Contents*: Direct access volumes only. This field is not used for tape volumes and should be recorded as blanks.
- *Processing*: Not used or verified. The EDGINERS and IEHINITT utility programs write blanks in this field.

6—Reserved (25 bytes)

- *Contents*: Reserved for possible future use—should be recorded as blanks. The EDGINERS and IEHINITT utility programs write blanks in this field. The IEHINITT utility program writes blanks in this field.

Historical Note: In certain levels of MVS/DFP* Version 3, position 36 may have been written as “P” to signify that the data was compacted via IDRC. The system no longer does this.

7—Owner Name and Address Code (10 bytes)

- *Contents*: Indicates a specific customer, person, installation, department, and so forth, to which the volume belongs. Any code or name is acceptable.
- *Processing*: Not used or verified. The EDGINERS and IEHINITT utility programs write the text specified by the user, and the open and EOVR routines write the text specified by the operator. If the code is less than 10 bytes, it is left-justified and the remainder of the field is padded with blanks. EDGINERS and IEHINITT allows EBCDIC alphanumeric and special characters.

8—Reserved (29 bytes)

- *Contents*: Reserved for possible future use—should be recorded as blanks.
- *Processing*: Not used or verified. The EDGINERS and IEHINITT utility programs write blanks in this field.

Format of IBM Standard Data Set Label 1 (HDR1/EOV1/EOF1)

IBM standard data set label 1 is 80 characters in length and describes the associated data set. The format is used for header labels (HDR1), end-of-volume trailer labels (EOV1), and end-of-data-set trailer labels (EOF1). Data set label 1 is always followed by data set label 2 when the system is writing labels. Data set label 1 is recorded in EBCDIC on 9-track tape units, or in BCDIC on 7-track tape units.

Figure 9 on page 26 shows the format of data set label 1. The shaded areas represent fields that the operating system writes in the label, but that are not used or verified during processing. The contents and processing of each field of the label are described below. The processing descriptions refer to the following system control blocks:

- Communication vector table (CVT)
- Data control block (DCB)
- Data extent block (DEB)
- Job file control block (JFCB)
- Unit control block (UCB).

Position	(Bytes)	Field Number and Name
1	(3)	1 Label Identifier
3		2 Label Number
4		
5	(17)	3 Data Set Identifier
21	(6)	4 Data Set Serial Number
22		
27	(4)	5 Volume Sequence Number
28		
31		
32	(4)	6 Data Set Sequence Number
35	(4)	7 Generation Number
36		
39	(2)	8 Version Number
40		
41		
42	(6)	9 Creation Date
47	(6)	10 Expiration Date
48		
53	(1)	11 Data Set Security
54		
55	(6)	12 Block Count, Low Order
60	(13)	13 System Code
61		
73	(3)	14 Reserved
74		
	(4)	15 Block Count, High Order
80		

Figure 9. Format of IBM Standard Data Set Label 1

1—Label Identifier (3 bytes)

- *Contents:* Three characters that identify the label are as follows:

HDR Header label (at the beginning of a data set)

- EOV Trailer label (at the end of a tape volume, when the data set continues on another volume)
- EOF Trailer label (at the end of a data set).

- *Processing:* The system checks this field to verify that the record is an IBM standard data set label.

For input data sets, the system checks the label identifier to determine whether data set processing is to be continued. When the system finds an EOV label, it performs volume switching. When the system finds an EOF label, it passes control to the user's end-of-data routine or continues processing with a concatenated data set.

If the DD statement specifies OPTCD=B for an input data set, the trailer label identifier (EOV or EOF) is not used to determine whether a volume switch is necessary. If more volumes are available, the system performs the switching. If no volumes are available, the system passes control to the user's end-of-data routine or continues processing with a concatenated data set.

When creating trailer labels, the EOV routine writes EOV in this field, and the close routine writes EOF.

2—Label Number (1 byte)

- *Contents:* The relative position of this label within a set of labels of the same type; it is always a 1 for data set label 1.
- *Processing:* Verified and written in conjunction with Field 1 to identify this label as HDR1, EOV1, or EOF1.

3—Data Set Identifier (17 bytes)

- *Contents:* The rightmost 17 bytes of the data set name (includes GxxxxVyy if the data set is part of a generation data group). If the data set name is less than 17 bytes, it is left-justified and the remainder of this field is padded with blanks. If the name contains embedded blanks or other special characters, you must enclose the name in apostrophes on the DD statement that requests this data set. *OS/390 MVS JCL User's Guide* lists the restrictions that apply to enclosing a data set name in apostrophes. The apostrophes do not appear in the data set identifier field.
- *Processing:* For input, this name is compared to the user-specified data set name found in the JFCB. This ensures that the correct data set is being processed.

For output, the data set name in the existing label is verified in conjunction with password protection to determine whether the existing data set can be overwritten. If protection is not specified, the data set name is not checked.

For input and output, when DFSMSrmm is used, the first file data set identifier is compared with the last 17 characters of the first file data set name recorded in the DFSMSrmm control data set. If there is no match, then the volume is rejected. This check ensures that the correct volume is mounted.

MVS can accept generation data group members written in VSE/ESA format. That is, on input, the operating system can verify the file name field on a VSE/ESA tape, excluding the generation and version number as a level of qualification, and verifies the generation and version number fields in addition to the file name field.

When creating labels for a new data set, the user-specified data set name is obtained from the JFCB and recorded in this field.

4—Data Set Serial Number (6 bytes)

- *Contents:* The volume serial number of the tape volume containing the data set. For multivolume data sets, this field contains the serial number of the first volume of the aggregate created at the same time. The serial number can be any 6 alphanumeric characters, typically numeric (000000 to 999999). The number may be from 1 to 6 characters, but, if fewer than 6 characters, the code must be left-justified and followed by blanks. Although all national characters, the hyphen, and other special characters are accepted when enclosed by apostrophes, their use is not recommended. This is because difficulties can occur in recognizing volume serial numbers when typewriter heads and print chains with nonalphanumeric characters are used.
- *Processing:* If this field has inconsistent values in the volumes in a volume set as described in Figure 6 on page 18 and Figure 7 on page 19, the system writes a warning message. This warns that one of the volumes might be the wrong volume. The message number is IEC709I. When creating labels, the serial number is obtained from the UCB and recorded in this field.

5—Volume Sequence Number (4 bytes)

- *Contents:* A number (0001 to 9999) that indicates the order of the volume within the multivolume group created at the same time. This number is always 0001 for a single volume data set.
- *Processing:* If this field does not have a value one greater than the previous volume in a volume set as described in Figure 6 on page 18 and Figure 7 on page 19, the system writes a warning message IEC709I. This warns the volumes might be out of order. When creating labels, the open routine writes 0001 in this field; the EOVS and close routines write a number one higher than on the previous volume.

6—Data Set Sequence Number (4 bytes)

- *Contents:* A number (0001 to 9999) that indicates the relative position of the data set within a multiple data set group. This number is always 0001 for a single data set organization.
- *Processing:* This number in the first HDR1 label on the tape is referred to when the open routine positions the tape. If this number in the first HDR1 label and the requested data set sequence number in the JFCB are both greater than 1, the logical data set sequence number in the UCB is set to the number in the label. Otherwise, the logical data set sequence number in the UCB is set to 1.

When creating labels, the open and close routines obtain the user-specified data set sequence number from the JFCB (a 0 is changed to 1). The EOVS routine propagates the number.

7—Generation Number (4 bytes)

- *Contents:* If the data set is part of a generation data group, this field contains a number from 0001 to 9999 indicating the absolute generation number. The first generation is recorded as 0001. If the data set is not part of a generation data group, this field contains blanks.

- *Processing:* Not used or verified. The generation number is available as part of the data set name in Field 3 of this label.

When creating labels, the system checks the JFCB to determine if the data set is part of a generation data group. If so, the generation number is obtained from the last part of the data set name in the JFCB. Otherwise, this field is recorded as blanks.

8—Version Number (2 bytes)

- *Contents:* If the data set is part of a generation data group, this field contains a number from 00 to 99 indicating the version number of the generation (the first version is recorded as 00). If the data set is not part of a generation data group, this field contains blanks.
- *Processing:* Not used or verified. The version number is available as part of the data set name in Field 3 of this label.

When creating labels, the system checks the JFCB to determine if the data set is part of a generation group. If so, the version number is obtained from the last part of the data set name in the JFCB. Otherwise, this field is recorded as blanks.

9—Creation Date (6 bytes)

- *Contents:* Date when the data set was created. The date is shown in the format cyyddd, where:

c = century (blank=19; 0=20; 1=21; etc.)
 yy = year (00-99)
 ddd = day (001-366)

Note: The century code gives the first two digits of the year, not the actual century. For example, a blank which translates into 19 indicates a year in the 1900s, not in the nineteenth century.

- *Processing:* Not verified. When the system creates labels, the date is obtained from the JFCB. If a data set is allocated via JCL, the creation date is the date when allocation begins for the job step responsible for creating the data set. If a data set is allocated dynamically, the creation date is the date when allocation begins for the data set.

10—Expiration Date (6 bytes)

- *Contents:* Year and day of the year the data set may be scratched or overwritten. The date is shown in the format cyyddd, where:

c = century (blank=19; 0=20; 1=21; etc.)
 yy = year (00-99)
 ddd = day (001-366)

Note: The century code gives the first two digits of the year, not the actual century. For example, a blank which translates into 19 indicates a year in the 1900s, not in the nineteenth century.

- *Processing:* For input, not used or verified. For output, the expiration date in the existing label is compared to the current date. If the date in the label is greater than the current date, the operator receives a message and is given the option of using the tape or mounting another. Other data sets that follow on the same volume expire on the same day.

If you enter 99365 or 99366, the system permanently retains your data sets, even after 1999. Do not use those dates as expiration dates. Use them as no-scratch dates only.

The expiration dates of scratch SMS-managed tape volumes, in both automatic and manual tape libraries are ignored. DFSMSrmm can override the expiration date or reject the volume without the operator's intervention.

When creating labels, the system obtains the expiration date from the JFCB. If you do not specify a retention period or expiration date, the expiration date is recorded as zeros, and the data set expires.

DFSMSrmm uses the expiration date to update the volume information in its control data set. For data sets without an expiration date, DFSMSrmm assigns one using the DFSMSrmm default retention period, but the label field is left as zeros.

11—Data Set Security (1 byte)

- *Contents:* A code number indicating the security status of the data set is as follows:

- | | |
|---|---|
| 0 | No password protection |
| 1 | Password protection. Additional identification of the data set is required before it can be read, written, or deleted. The password is ignored if the volume is RACF defined. |
| 3 | Password protection. Additional identification of the data set is required before it can be written or deleted. The password is ignored if the volume is RACF defined. |

- *Processing:* For input, the system inspects this field on a single volume data set, on each concatenated data set, and on each volume of a multivolume data set. If protection is specified in this field, the system verifies the password furnished by the operator (or the user if under TSO) and sets a security indicator in the JFCB.

For output, the system inspects this field in the existing HDR1 label. If security is specified, the existing data set cannot be overwritten until the system verifies the password and the data set name in Field 3 of this label. If you specify a data set name different from the one in Field 3, and the data set is the first one on the first or only volume, the operator is requested to remove the tape and mount a scratch tape, even though you requested a specific volume. If the data set is not the first one on the volume or this is not the first volume of a multivolume data set, the system abnormally terminates the task.

When the second or later data set on a volume is created, and there is no HDR1 label with which to determine security protection, open reads the EOF1 label of the preceding data set on the volume. The data set security level in the EOF1 label must match the security level requested for the new data set. If they are not equal, the system abnormally terminates the task. If the security levels are equal and indicate no security protection, open processing continues. If security protection is indicated, the system requests a password and verifies that the password furnished by the operator allows access to the data set name in the EOF1 label, the preceding data set. It is important to note that, in this case, only the 17-byte data set name in the EOF1 label is available. Therefore,

either the data set name must be 17 or fewer characters in length, or the last significant 17 characters of the full data set name must be entered in the PASSWORD data set. It is recommended that password-protected tape data sets limit their data set names to 17 or fewer characters, or that the last 17 characters of the data set name be entered in the password data set together with the full data set name.

When the system creates labels, the user's request for security is determined from the indicator in the JFCB.

12—Block Count, Low Order (6 bytes)

- *Contents:* This field in the trailer label shows the low order six digits of the number of data blocks in the data set on the current volume. This field in the header label always contains zeros (000000).
- *Processing:* The DCB count is increased as the data set is read. The final DCB count is compared with the count in fields 12 and 15 in the trailer label at end of data or end of volume. If the counts do not agree, a user exit entry in the DCB exit list determines whether processing will continue or abnormally terminate. A block count discrepancy causes processing to abnormally terminate if the appropriate user exit entry is not provided.

For read backward, the verification process is reversed. The trailer label count is recorded in the DCB and decreased as the data set is read. The final DCB count should be zero, which equals the count in the header label.

When the system creates labels, the block count in the header label is set to zeros. The block count in the trailer label is obtained from the DCB.

If the data set was created with an EXCP DCB that had no device-dependent section, the block count is written as zero and is not verified.

13—System Code (13 bytes)

- *Contents:* A unique code that identifies the system: 'IBM OS/VS 370'.
- *Processing:* Not used or verified.

14—Reserved (3 bytes)

- *Contents:* Reserved for possible future use—contains blanks.
- *Processing:* Not used or verified. When creating labels, the system writes blanks in this field.

15—Block Count, High Order (4 bytes)

- *Contents:* High order four digits of a ten-digit block count. This field in a header label always contains blanks. If the data set contains more than 999999 blocks on the volume, this field contains up to four higher order digits. Of these four digits, the system changes high order zeroes to blanks when creating the trailer label.
- *Processing:* See field 12.

Format of IBM Standard Data Set Label 2 (HDR2/EOV2/EOF2)

IBM standard data set label 2 always follows data set label 1 and contains additional information about the associated data set. The format is used for header labels (HDR2), end-of-volume trailer labels (EOV2), and end-of-data-set trailer labels (EOF2). The label is 80 characters in length. It is recorded in EBCDIC on 9-track tape units and on cartridges, or in BCDIC on 7-track tape units. Tapes created on other operating systems such as VSE might not contain header label 2.

Figure 10 on page 33 shows the format of data set label 2. The shaded areas represent fields that the operating system writes in the label, but that are not used or verified during processing. The processing descriptions refer to the following system control blocks:

- Data control block (DCB)
- Job file control block (JFCB)
- Unit control block (UCB).

Position	(Bytes)	Field Number and Name
1	(3)	1 Label Identifier
3		
4	(1)	2 Label Number
5	(1)	3 Record Format
6	(5)	4 Block Length
10		
11	(5)	5 Record Length
15		
16	(1)	6 Tape Density
17	(1)	7 Data Set Position
18	(17)	8 Job/Job Step Identification
34		
35	(2)	9 Tape Recording Technique
36		
37	(1)	10 Control Character
38	(1)	11 Reserved
39	(1)	12 Block Attribute
40	(2)	13 Reserved
41		
42	(6)	14 Device Serial Number
47		

Figure 10. Format of IBM Standard Data Set Label 2

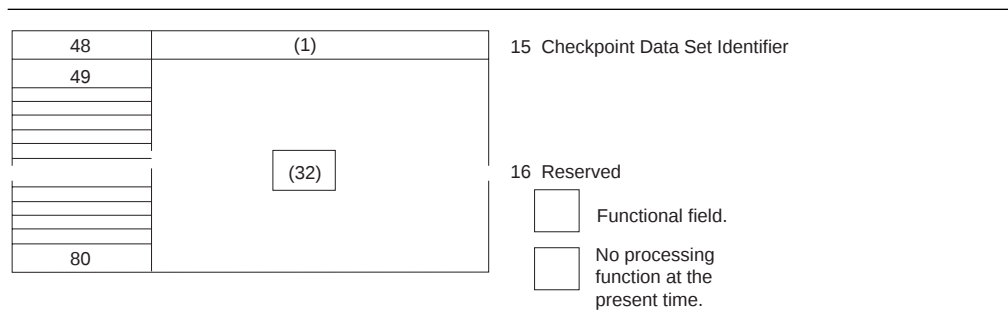


Figure 11. Format of IBM Standard Data Set Label 2 Continue

1—Label Identifier (3 bytes)

- *Contents:* Three characters that identify the label are as follows:

HDR	Header label (at the beginning of a data set)
EOV	Trailer label (at the end of a tape volume, when the data set continues on another volume)
EOF	Trailer label (at the end of a data set).

- *Processing:* The system checks this field to verify that the record is an IBM standard data set label.

For input data sets, the system checks the label identifier to determine whether data set processing is to be continued. When the system finds an EOV label, it performs volume switching. When the system finds an EOF label, it passes control to the user's end-of-data routine or continues processing with a concatenated data set.

If the DD statement specifies OPTCD=B for an input data set, the system accepts either EOV or EOF as the trailer label identifier, and the identifier is not used to determine whether a volume switch is necessary. If more volumes are available, the system performs the switching. If no volumes are available, the system passes control to the user's end-of-data routine or continues processing with a concatenated data set.

When creating trailer labels, the EOV routine writes EOV in this field, and the close routine writes EOF.

2—Label Number (1 byte)

- *Contents:* The relative position of this label within a set of labels of the same type; it is always a 2 for data set label 2.
- *Processing:* Verified and written in conjunction with Field 1 to identify this label as HDR2, EOV2, or EOF2.

3—Record Format (1 byte)

- *Contents:* An alphabetic character that indicates the format of the records in the associated data set:

F	Fixed length
V	Variable length
U	Undefined length.

- *Processing:* For input, the OPEN routine obtains the record format from this label, converts it to a bit code, and records it in the JFCB (if the JFCB field is zero). Then the record format in the JFCB is recorded in the DCB (if the DCB field is zero).

Note: This is a merging process in which existing specifications in the JFCB and DCB cannot be overridden.

When creating labels, the system's OPEN and EOVS routines do a reverse merge. The record format in the DCB overrides the record format in the JFCB, and the updated JFCB provides the information for the label.

This merging process is explained and illustrated in Figure 2 on page 4.

4—Block Length (5 bytes)

- *Contents:* A number up to 32760 that indicates the block length, in bytes. Interpretation of the number depends on the associated record format in Field 3, as follows:

Format F	Maximum block length (must be a multiple of the logical record length in Field 5). If field 12 contains a blank, then this field is also the minimum block length.
Format V	Maximum block length (including the 4-byte length field in the blocks)
Format U	Maximum block length.

The system can determine the optimum block size when creating tape data sets. For more information see "System-Determined Block Size," *DFSMS/MVS Using Data Sets*.

On a tape written on OS/400, the first byte of this field can contain a question mark. OS/390 does not currently accept such a tape.

- *Processing:* The number in the label is converted to binary and merged with appropriate fields in the JFCB and DCB. The merging process is the same as that for the record format code in Field 3 of this label.

5—Record Length (5 bytes)

- *Contents:* A number that indicates the record length, in bytes. Interpretation of the number depends on the associated record format in Field 3, as follows:

Format F	Logical record length
Format V	Maximum logical record length (including the 4-byte length field in the records)
Format U	Zeros.

- *Processing:* The number in the label is converted to binary code and merged with the appropriate fields in the JFCB and DCB. The merging process is the same as for the record format code in Field 3 of this label.

6—Tape Density (1 byte)

- *Contents:* A code indicating the recording density of the tape; the code is equivalent to the DEN parameter value on the DD statement. For the DEN parameter values, refer to "Tape Characteristics" on page 12. For non-cartridge tapes, when density is not specified in the DCB or JCL DD statement, this field will be set to the highest density applicable to the drive (default).

Note: Specifying DEN=0 for a 7-track 3420 results in 556 bits-per-inch recording, but corresponding messages and tape labels indicate 200 bits-per-inch recording density.

For tape cartridges, this field contains a character 0.

- *Processing:* Not used or verified. When the system creates labels, the information for this field is obtained from the JFCB.

7—Data Set Position (1 byte)

- *Contents:* A code indicating a volume switch is as follows:

0	No volume switch has occurred
1	A volume switch previously occurred.

- *Processing:* Not used or verified. When creating labels, the open routine writes 0 in this field, and the EOVS routine writes 1. The close routine determines which code to write by comparing the first volume serial number in the JFCB to the number in the UCB. It writes 0 if the numbers are equal, and 1 if they are not equal.

8—Job/Job Step Identification (17 bytes)

- *Contents:* Identification of the job and job step that created the data set. For MOD processing, EOF2 contains the name of the job and job step that extended it. The first 8 bytes contain the name of the job, the ninth byte is a slash (/), and the final 8 bytes contain the name of the job step.
- *Processing:* Not used or verified. When the system creates labels, it sets this field as described above.

9—Tape Recording Technique (2 bytes)

- *Contents:* A code or blanks indicating the tape recording technique used to create the data set.

For 7-track tapes the values are:

Tb	Odd parity with translation
Cb	Odd parity with conversion
Eb	Even parity with no translation
ET	Even parity with translation
bb	Odd parity with no translation or conversion.

The only technique available for 9-track tape is odd parity with no translation.

For a magnetic tape subsystem with Improved Data Recording Capability, the values are:

Pb	Record data in compacted format
bb	Record data in standard, uncompact format

For other tapes, this field is recorded as blanks.

- *Processing:* For 7-track tapes and cartridges written by a magnetic tape subsystem with Improved Data Recording Capability, the specification in the label is converted to a bit code and merged with the appropriate fields of the JFCB and DCB. The merging process is the same as that for the record format code in Field 3 of this label.

10—Control Character (1 byte)

- *Contents:* A printer control code indicating whether a control character set was used to create the data set and the type of control characters used:

A	Contains ANSI control characters
M	Contains machine control characters
(blank)	Contains no control characters.

- *Processing:* The specification in the label is converted to a bit code merged to the appropriate fields of the JFCB and DCB. The merging process is the same as that for the record format code in Field 3 of this label.

11—Reserved (1 byte)

- *Contents:* Reserved for possible future use (recorded as blanks).
- *Processing:* Not used or verified. When creating labels, the system writes blanks in this field.

12—Block Attribute (1 byte)

- *Contents:* A code indicating the block attribute used to create the data set:

B	Blocked records
S	Spanned records, if the record format byte = 'V'
S	Standard records, if the record format byte = 'F'
R	Blocked and spanned records, if the record format byte = 'V'
R	Blocked and standard records, if the record format byte = 'F'
(blank)	Records that are not blocked and not spanned, or records that are not blocked and not standard.

- *Processing:* The specification in the label is converted to a bit code and merged with the appropriate fields of the JFCB and DCB. The merging process is the same as for the record format code in Field 3 of this label.

13—Reserved (3 bytes)

- *Contents:* Reserved for possible future use (recorded as blanks).
- *Processing:* Not used or verified. When creating labels, the system writes blanks in this field.

14—Device Serial Number (6 bytes)

|

- *Contents:* For tape reels used by the 3420, byte 43 contains the model number, and bytes 44 through 47 contain the last 4 digits of the serial number of the creating tape unit.

For tape cartridges written by IBM magnetic tape subsystems, bytes 43 through 46 contain the last 4 digits of the serial number of the control unit, and byte 47 contains the device address.

For IBM 3590 High Performance Cartridge Tapes written by IBM magnetic tape subsystems, bytes 42 through 47 contain the 6 digit device serial number.

Note: The serial numbers in the header and trailer labels may not be the same if the data set was extended, or if the system swapped tape units while the data set was being created.

- *Processing:* A unique number to identify the recording unit is read off the tape during open processing, converted into binary, and inserted into the UCBCTD field in the UCB tape extension.

15—Checkpoint Data Set Identifier (1 byte)

- *Contents:* This byte contains the character C if the data set is a secure checkpoint data set; the byte is blank if the data set is not a secure checkpoint data set.
- *Processing:* This field is examined by open/close/EOV. If it finds the data set is a checkpoint data set, it performs the following security operations:
 - Verifies (via messages to the operator) that the data set is a secure checkpoint data set.
 - Determines whether the user is authorized. If the user is unauthorized, open/close/EOV does not allow access to the checkpoint data set directly, although it will allow the taking of checkpoints and performing restarts.

16—Reserved (32 bytes)

- *Contents:* Reserved for possible future use (recorded as blanks).
- *Processing:* Not used or verified. When creating labels, the system writes blanks in this field.

Format of IBM Standard User Label (UHL1-UHL8 or UTL1-UTL8)

IBM standard user labels contain user-specified information about the associated data set. User labels are optional within the standard label groups.

Figure 12 on page 39 shows the format of user labels. The format is used for user header labels (UHL1-UHL8) and user trailer labels (UTL1-UTL8). The labels are 80 characters in length. They are recorded in the density specified via JCL. Seven-track tape records are in BCDIC. The contents and processing of each field of the label are described below.

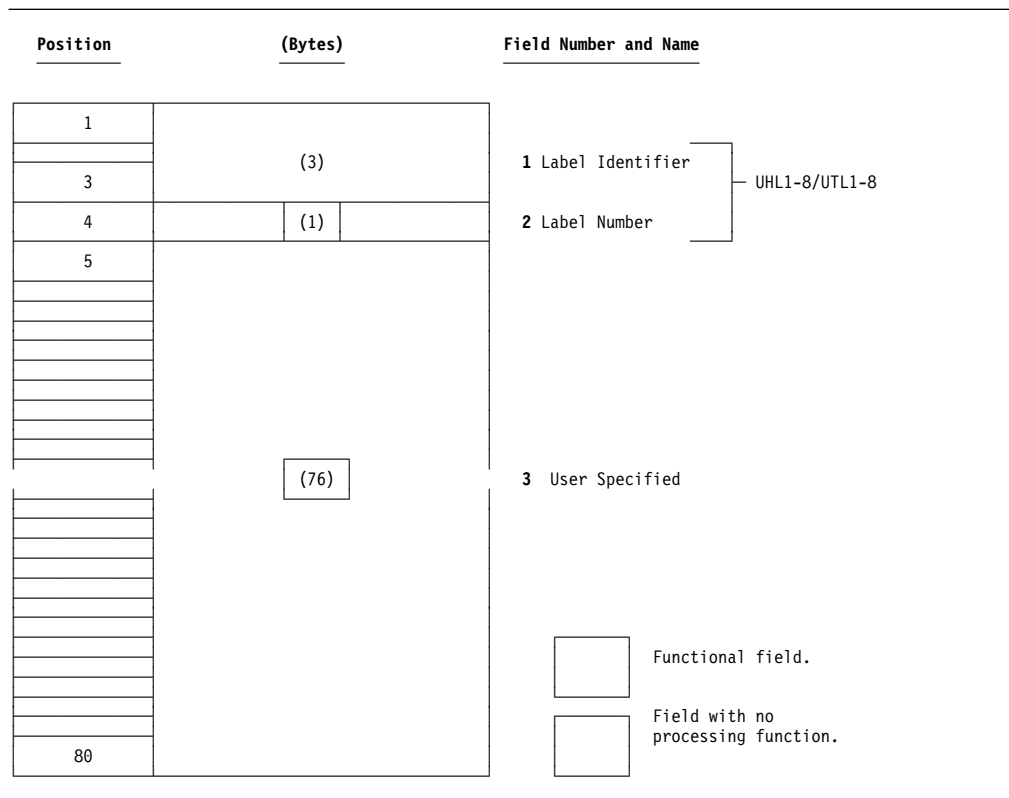


Figure 12. Format of User Label

- 1—Label Identifier (3 bytes)**
 - Contents:* Three characters that identify the label are as follows:

UHL User header label (at the beginning of a data set)
 UTL User trailer label (at the end-of-volume or end-of-data-set).
 - Processing:* This field is read to verify that the record is a user label. The system accepts either UHL or UTL.
- 2—Label Number (1 byte)**
 - Contents:* The relative position of this label within a set of labels of the same type; it can be a number from 1 to 8.
 - Processing:* This field is read to ensure that no more than eight user labels are processed. This field is read in conjunction with Field 1.
- 3—User Specified (76 bytes)**
 - Contents:* Specified by the user.
 - Processing:* Specified in the DCB exit list.

IBM Standard Label Processing

Label processing is handled by the open/close/end-of-volume routines. This processing consists of the following basic functions:

- Checking the labels on input tapes to ensure that the correct volume is mounted, and to identify, describe, and protect the data set being processed
- Checking the existing labels on output tapes to ensure that the correct volume is mounted, and to prevent overwriting of vital data
- Creating and writing new labels on output tapes
- For volumes in an automated tape library, comparing the external volume serial number to the magnetically recorded internal label.

These processing functions are summarized in Figure 13. The table shows the specific labels that are processed for each function and which routines perform the functions.

Figure 13 (Page 1 of 2). IBM Standard Label Processing by Open/Close/EOV Routines

Processing	Volume Label VOL1	Header Labels ¹			Trailer Labels ¹		
		HDR1	HDR2	UHL1-8	EOF1 or EOVS	EOF2 or EOVS	UTL1-8
First or only volume ²							
Checks labels on input tape	Open	Open	Open	Open	EOV	Bypassed	EOV
Checks existing labels on output tape before overwriting	Open	Open	Open	Not read	Not read	Not read	Open ³
Writes new labels on output tape	Open or user ⁴	Open	Open	Open	Close or EOVS	Close or EOVS	Close or EOVS
Compares the volume serial number on the external volume label to the serial number in the tape volume label ⁶	Open	Open	N/A ⁷	N/A ⁷	N/A ⁷	N/A ⁷	N/A ⁷
Second or subsequent volumes ⁵							
Checks labels on input tape.	EOV	EOV	Bypassed	EOV	EOV	Bypassed	EOV
Checks existing labels on output tape before overwriting.	EOV	EOV	Not read	Not read	Not read	Not read	Not read
Writes new labels on output tape.	EOV or user ⁴	EOV	EOV	EOV	Close or EOVS	Close or EOVS	Close or EOVS
Compares the volume serial number on the external volume label to the serial number in the tape volume label ⁶	EOV	N/A ⁷	N/A ⁷	N/A ⁷	N/A ⁷	N/A ⁷	N/A ⁷

Figure 13 (Page 2 of 2). IBM Standard Label Processing by Open/Close/EOV Routines

Processing	Volume Label VOL1	Header Labels ¹			Trailer Labels ¹		
		HDR1	HDR2	UHL1-8	EOF1 or EOV1	EOF2 or EOV2	UTL1-8

Notes:

1. For read backward operations, the action on header and trailer labels is reversed.
2. Includes the first volume of concatenated data sets with unlike characteristics when the user program has turned on the DCB unlike attributes bit (DCBOFPPC). See "Concatenating Unlike Datasets" in *DFSMS/MVS Using Data Sets*.
3. If DISP=MOD is specified on the DD statement, or the OPEN macro specifies EXTEND or OUTINX, then the open routine positions the tape at the end of the existing data set and allows an input user trailer label routine to process user trailer labels (prior to overwriting the existing labels).
4. User can create the label with the EDGINERS or IEHINITT utility programs, or a user program. Subsequently, the label may be rewritten by the open and EOV routines.
5. Includes the first volume of concatenated data sets when the user has not turned on the DCB unlike attributes bit.
6. Applies only to Tape Library Dataserver.
7. Not Available.

When a data set is opened for input, the volume label and HDR1 are processed. HDR2 is processed if it exists. For an input end-of-data condition, the trailer labels are processed, unless deferred user input trailer label processing is specified in the DCB exit list. For an input end-of-volume condition, the trailer labels on the current volume are processed, and then the volume label and header labels on the next volume are processed. When the FEOV macro is issued for an input tape, the trailer labels on the current volume are not processed, but the volume labels and header labels on the next volume are processed. No label processing is performed when an input data set is closed, unless deferred user input trailer label processing is specified in the DCB. If deferred user input trailer label processing is specified, the processing otherwise performed for an input end-of-data condition is performed when an input data set is closed.

When a data set is opened for output, the existing volume label and HDR1 label are checked, and a new volume label and new header labels are written. For an output end-of-volume condition (including FEOV), trailer labels are written on the current volume, the existing volume and header labels on the next volume are checked, and then a new volume label and new header labels are written on the next volume. When an output data set is closed, trailer labels are written.

Creating a Volume Label

The IBM standard volume label (VOL1) is usually written by a utility program when the volume is first received at the installation. At that time, a permanent volume serial number is assigned to the volume, physically posted on the volume, and recorded in the VOL1 label.

You can use the EDGINERS or IEHINITT utility programs to create IBM standard volume labels. EDGINERS and IEHINITT initialize the tape by writing in the following order:

1. A volume label (VOL1) with the volume serial number and owner identification that you specify. You cannot specify any other fields of the VOL1 label.
2. A dummy header label (HDR1 followed by 76 EBCDIC zeros).
3. A tapemark.

The IEHINITT utility program can write a volume label on a labeled, unlabeled, or blank tape; it makes no checks to see what data, if any, previously existed on the tape. Therefore, IEHINITT does not check for password or RACF security protection. It does not create, modify, or delete RACF profiles of RACF-defined volumes. Detailed procedures for using the program are described in *DFSMS/MVS Utilities*.

The EDGINERS utility program verifies that the volume is mounted before writing a volume label on a labeled, unlabeled, or blank tape. EDGINERS does not check password or RACF security protection, but it verifies whether the volume is defined to DFSMSrmm. DFSMSrmm must know that the volume needs to be labelled. If the labelled volume is undefined, then DFSMSrmm defines it to DFSMSrmm and can create RACF volume security protection. Detailed procedures for using the program are described in *DFSMS/MVS DFSMSrmm Implementation and Customization Guide*.

Methods other than the EDGINERS or IEHINITT utility programs can be used to write volume labels. You can use a card-to-tape program, or you can replace the IBM-supplied volume label editor routine with one that writes volume labels. If you use an editor routine to write the volume label, some data or a tapemark should already exist on the tape. See “IBM Standard Volume Label Verification and Editor Routines” on page 43 for more information. Otherwise, the system might read through the entire reel of blank tape looking for a label.

The VOL1 label is rewritten by the open or EOVR routine if **all** the following conditions are met. It is not necessary for a conflict to occur in order for the VOL1 label to be rewritten:

- If a density or label type conflict occurred, then permission must be granted by the installation's label anomaly exit, the installation's label editor routine or the system operator.
- OUTPUT, OUTIN, or OUTINX is specified in the OPEN macro.
- The tape is positioned to the first data set on the volume.
- Either of the following two conditions are true:
 - The data set is not password protected, or
 - The volume is RACF protected, the system-wide RACF tape protection option has been specified, and the user is ALTER authorized.

All VOL1 labels that can be successfully read under these conditions are rewritten.

If you request a standard labeled (SL, SUL) output volume and the tape that you are allocated is recorded in the wrong density and cannot be read, the VOL1 label is rewritten in the density that you specify. This facility allows you to make nonspecific requests; you do not need to specify a volume serial number in your DD statement for output tapes. The operator can mount any scratch volume to answer your request. However, if the system-wide RACF tape protection option has been specified, the volume is rejected, because it cannot be verified that it is not a RACF-protected volume.

If you make a nonspecific output volume request for a standard labeled (SL, SUL) tape and the mounted volume is an NL or NSL labeled tape, the open or the EOVR routine creates a volume label (VOL1). A message is sent to the console operator

requesting serial number and owner information. DFSMSrmm can optionally prevent the creation or the destruction of standard labels.

IBM Standard Volume Label Verification and Editor Routines

If you specify that an input or output tape has a standard label, the operating system checks for the standard volume label at the beginning of the tape.

Because of conflicting label types or tape characteristics, various error conditions can occur during the verification of the first record. Under some error conditions, the tape is accepted for use. Under other error conditions, the tape is not accepted and the system issues another mount message. For certain other error conditions, the system gives control to a label anomaly exit routine and to *avolume label editor routine*; your installation can use IBM-supplied routines, or it can supply its own routines.

The IBM-supplied volume label editor routines determine the discrepancies between the requested tape and the mounted tape, and if necessary, pass control to the appropriate system routine to create or destroy labels. If DFSMSrmm is controlling the volume, the DFSMSrmm label editor routine controls the creation or destruction or standard labels using RACF profiles in the FACILITIES class. Installation-supplied routines can perform other functions.

IBM recommends using tape management exit routines instead of the volume label editor routines.

For detailed information about these and other available exit routines, see *DFSMS/MVS Installation Exits*.

Protecting Data

Data on IBM standard volumes can be protected by either RACF or data set password protection.

IBM recommends RACF instead of data set password protection. For more information see, *DFSMS/MVS DFSMSdfp Advanced Services*.

Note: All checking for authorization will be bypassed if security processing is suppressed. This can occur, for example, when the program properties table is marked to suppress security checking. Only the system programmer can update the program properties table. For information about the program properties table, see *OS/390 MVS Initialization and Tuning Guide*.

RACF allows you to control access to either the tape volume(s) or the individual data sets on the tape. RACF protection at the volume level overrides RACF protection at the data set level. See *OS/390 Security Server (RACF) Security Administrator's Guide* for information on how to activate these levels of RACF protection, and how they interact with each other and with your own tape management system, if applicable.

DFSMSrmm supports RACF protection, but not password protection. For more information about DFSMSrmm and RACF, see *DFSMS/MVS DFSMSrmm Implementation and Customization Guide*.

The following principles apply to RACF protection at the volume level:

- ALTER access authority is required to create or destroy the tape volume label.
- READ access authority is required to open the volume for input (open options INPUT or RDBACK).
- UPDATE access authority is required to open the volume for output (open options OUTPUT, EXTEND, INOUT, OUTIN, or OUTINX).

If the tape volume is defined to RACF, the user has UPDATE access authority, and PROTECT=YES has not been specified in the JCL, the user can open the volume to read or write.

If the tape volume is defined to RACF and the user has UPDATE authority, and PROTECT=YES has been specified in the JCL, and the tape is not a RACF scratch volume, the request fails.

If the tape volume is defined to RACF and the user has READ but not UPDATE access authority, or if the user has UPDATE access but PROTECT=YES has been specified in the JCL and the volume is a RACF scratch tape volume, the system does not grant the user access to read until it has ensured that the user will not be able to write on the tape. The user cannot access the volume until one of the following conditions is met:

1. **Hardware Protection.** If the write-enable ring has been removed from the tape reel or the write-protect tab has been set to disable writing on the tape cartridge, the tape volume cannot be written on by any user, so the system safely permits the user to access the tape to read. This hardware protection cannot be circumvented by software.
2. **Logical write-protection.** If the write-protect tab on an IBM magnetic tape cartridge is set to enable writing, the system issues a hardware command to prevent writing on that cartridge. If the command succeeds, the system safely allows the user to access the tape to read. An unauthorized program cannot bypass this combination of hardware and software protection.
3. **IEC.TAPERING.** Your installation may choose to depend on a tape management system to prevent overwriting unexpired data on tapes. Typically, a tape management system only allows volumes with no unexpired data to be opened for output. DFSMSrmm provides facilities to prevent accidental overwriting of non-scratch volumes. The IEC.TAPERING support facilitates the operation of tape management systems because it allows all volumes to remain write-enabled (by the ring in the volume or the switch on the cartridge), eliminating the need for further operator intervention.

If the write ring or cartridge tab is set to enable writing, the system checks if the user is authorized for read to the IEC.TAPERING profile in the RACF FACILITY class. If the user does have this authority, the system grants the user access.

Attention If you use the IEC.TAPERING support to allow users to read from tapes that are enabled for writing (when the users are only authorized to read), the system software cannot prevent knowledgeable users from also writing on any files on the tapes.

4. **Operator Intervention.** If none of the preceding conditions are met, the system requires the operator to intervene and prevent writing on the volume. The system demounts the tape and issues a message asking the operator to remove the write-enable ring from the tape reel or change the switch on the

tape cartridge. After the operator remounts the tape, the system continues to protect the volume from unauthorized writing by repeating the preceding checks, beginning with the check for hardware protection, until one of the conditions is met.

If the tape volume is not defined to RACF, access is granted and processing continues. For an overview of RACF protection for tape volumes, see *OS/390 Security Server (RACF) Security Administrator's Guide*. For information on how DFSMSrmm can help you manage RACF security for your tape volumes, see *DFSMS/MVS DFSMSrmm Implementation and Customization Guide*.

Data set password protection is described in *DFSMS/MVS DFSMSdfp Advanced Services*.

Opening an Input Data Set

If IBM standard labels are specified, the first record on the input tape must be an IBM standard volume label (VOL1). At the time the data set is opened, the system checks the first record on the tape to determine whether the record is 80 bytes long and contains the identifier VOL1 in the first 4 bytes. The various error conditions that can occur during verification of the first record are explained in "IBM Standard Volume Label Verification and Editor Routines" on page 43.

Volume Serial Number

The system uses the VOL1 label to ensure that the correct tape is mounted. The volume serial number in the label is compared to the volume serial number that you specify. You can specify the serial number either directly in the DD statement or indirectly through the catalog facility. Serial numbers are required when the processing method is INPUT, INOUT, RDBACK, EXTEND or OUTINX.

Serial numbers are also required when the processing option is OUTPUT or OUTIN with DISP=MOD.

If the volume serial number is correct, the system resets a mount switch in the unit control block to indicate that volume mounting is verified. The switch is initially set when the mount message is issued to the operator. If the serial number is not correct, the system rejects the tape and issues another mount message.

Positioning the Volume to the Data Set

When the volume is mounted and verified, the system positions the tape in front of the header label group of the data set to be processed. Usually, there is only one data set on the volume, and the header label group immediately follows the volume label.

To retrieve a data set on a single volume of tape that contains multiple data sets, you specify a data set sequence number in the LABEL parameter of the DD statement, unless the data set is cataloged. For a cataloged data set, you need not specify a data set sequence number because the system obtains the number from the catalog along with the volume serial number.

- The sequence number can be from 1 to 9999, with 1 representing the first data set on the volume. If you specify a sequence number higher than the number

of data sets on the volume, your task will be abnormally terminated or, if the volume ends with EOVS labels, the open routine will switch to the next volume.

- If the data set is not cataloged and you do not specify a sequence number, or you specify 0, the system assumes that the data set is the first in sequence on the volume.

To position the tape, the system uses the requested data set sequence number shown in the JFCB and the data set sequence number shown in the first HDR1 label on the tape, and maintains a logical data set sequence number in the unit control block (UCB). The number in the UCB represents the current position of the tape and is maintained as follows:

1. When a tape is first mounted, the data set sequence number in the UCB is 0.
2. When a data set is opened, the open routine sets the data set sequence number in the UCB to 1. The exceptions are:
 - If the tape is still positioned from previous processing, such as for a LEAVE request, the open routine does not reset the number in the UCB.
 - If the data set sequence number in the JFCB and the data set sequence number in the first HDR1 label on the tape are both greater than one, the open routine sets the data set sequence number in the UCB to the value of the number in the first HDR1 label. (The data set sequence number in the first HDR1 label may be greater than one when the volume is part of a multiple-data-set/multiple-volume aggregate.)
 - When the processing method is INPUT, INOUT, or OUTPUT to the start of a data set on a multiple file tape, the open routine starts with the first volume, unless a volume sequence number is specified. If the volume ends with EOVS labels before the desired file sequence number, the open routine will switch to the next volume and permanently update the volume sequence number so that the next open to this data set will start with the correct volume.
 - When the processing method is RDBACK, the open routine speeds up finding the end of the data set by starting with the last volume specified, unless a volume sequence number is specified.

Note: The use of the DCB parameter in the JCL causes a specific volume sequence number of one. If the data set is not yet present on the last volume specified, the open routine can recover, if the file sequence number is 1, by backing up volumes. It detects that the data set is not present if the *dsname* is invalid, the tape starts at a file sequence number greater than 1, or the VOL label is followed by a tapemark.

3. The data set sequence number in the UCB is compared to the requested data set sequence number in the JFCB. If they are equal, the tape is already positioned at the requested data set. If they are not equal, the open routine adjusts the data set sequence number in the UCB as the tape is positioned past each data set, until the number in the UCB equals the number in the JFCB.
4. When multiple tape units are used, and a volume switch causes processing to be continued on a volume on a different unit, the EOVS routine copies the data set sequence number from the previous UCB to the current UCB.
5. If the data set is not open or has been closed, the UCBFSC data set sequence number field of the UCB will be set to X'0000' if:

- The data set was never opened.
- CLOSE (,REWIND) was specified.
- CLOSE (,REREAD) and LABEL=1 was specified.
- CLOSE (,DISP) was specified or defaulted, and DISP=(,PASS) was not specified on the JCL.

Otherwise, the UCBFSCT field of the UCB will have a value one greater than the value specified on the LABEL parameter of the JCL.

6. If the job terminates abnormally while a tape data set is open, the data set will be closed and the tape will be positioned as when CLOSE (,LEAVE) is specified. That is, the UCBFSCT field of the UCB will have a value one greater than that specified on the LABEL= parameter of the JCL.

There are several instances in which a volume is repositioned to the next (or previous) tapemark during open. This is usually done by reading data but suppressing data transmission to storage until a tapemark is found, but can be done by I/O spacing commands (for example, BACKSPACE FILE). To reduce the chance of an “unexpected record” condition (613-0C), the first method is preferred over the spacing commands. In the event of a 613-08 or 613-0C abend, a tape positioning installation exit (IFG0199I) is given control to allow recovery. For more information about the “data management abend installation” exit, see *DFSMS/MVS Installation Exits*.

Only one data set on a tape volume may be open at any given time. An attempt to begin processing a second data set on the same volume results in abnormal termination.

When the tape is positioned to the data set header label group of the first data set or the requested data set, the system checks the label identification. Processing is abnormally terminated if the HDR1 identifier is not found.

Processing the HDR1 Label

Verify Data Set Name

To ensure that the correct data set is being opened, the system compares the data set name shown in the HDR1 label of the requested data set to the data set name specified by the user in the DD statement. This comparison is made on only the 17 least significant (rightmost) characters of the data set name (including 8 characters for the generation and version numbers if the data set is part of a generation data group). DFSMSrmm checks the full 44-character data set name if the previous data set name information is known to DFSMSrmm.

Processing is abnormally terminated if the comparison shows an incorrect data set name.

Open/EOV User Exit for Security Verification

For authorized program facility (APF) programs for which the program property “bypass password (and RACF) checking” is active, a user exit is provided for verifying that a tape selected by Open or EOVS should be used, and whether certain security checks may be bypassed. For more information on the “Open/EOV volume security and verification” exit, see *DFSMS/MVS Using Data Sets*.

Password Protection

Password protection is not as strong as RACF protection. In each of the following situations, the system ignores any password protection for a tape and uses RACF protection:

- The volume is RACF protected
- The volume is in an automated tape library
- The volume is in a manual tape library

If RACF or an equivalent is not running, then the volume has no security protection.

Expiration Date

The expiration date shown in the HDR1 label is not verified for input data sets, unless the processing method is INOUT. For INOUT, if the expiration date has not been reached, the system notifies the operator and asks for confirmation of the use of the tape. If confirmation is not received, processing is abnormally terminated. If you override the INOUT specification by coding LABEL=(,,IN) on the DD statement, the expiration date is not verified. Automatic handling of expiration date protected tapes is provided by DFSMSrmm functions.

Block Count

The block count shown in the HDR1 label is always 0 (EBCDIC or BCDIC). This 0 is recorded in the data control block (in binary) and increased during processing for comparison to the block count shown in the trailer label (EOV1 or EOF1).

For reading backward, the block count shown in the trailer label (EOV1 or EOF1) is recorded in the data control block and decreased during processing for comparison to the 0 block count in the HDR1 label.

The block count is verified at end-of-data or end-of-volume. For more information, see “Verifying Block Count” on page 11 and “Verifying Block Count in Label When Reading” on page 50.

Processing the HDR2 Label

Data Set Characteristics

The HDR2 label immediately follows the HDR1 label. The system uses the HDR2 label to determine certain data set characteristics, if these characteristics are not otherwise specified by the user. The characteristics that can be obtained from the HDR2 label are:

- Record format
- Block length
- Logical record length
- Tape recording technique
- Type of control characters

The above information is obtained from the label and recorded in the job file control block (JFCB) and the data control block, provided that the appropriate fields in these control blocks contain zeros. The label information cannot override any characteristics previously specified in the processing program or the DD statement. For an explanation and illustration of this merging process, see “Completing the Data Control Block” on page 3.

Positioning After Header Label 2

Unless user header labels are to be processed, the system positions the tape past the tapemark immediately after processing the HDR2 label. All labels that follow the HDR2 label are bypassed, and the tape is positioned at the first data set record.

Note: If the HDR2 label is missing, processing will continue with the assumption that it is not needed. If the JFCB/DCB merge function needs it because of missing data set characteristics, the OPEN will abnormally terminate.

User Header Labels

Up to eight user header labels (UHL1 to UHL8) may follow the HDR2 label. To make the user header labels available to your program, SUL must be coded on the DD statement and the address of an input user header label routine must be specified in the DCB exit list. If you omit one of these parameters, the system positions the tape past the tapemark immediately after processing the HDR2 label.

Read Backward

For the read backward (RDBACK) processing method, the system uses the data set's trailer labels as header labels, and vice versa. Each label group is read in the normal sequence; that is, EOF1 is read before EOF2, and so forth. The data records, however, are read in reverse sequence.

Multivolume data sets can be read backward. Concatenated data sets, 7-track tape with data conversion, and format-V (variable-length) records cannot be read backward.

End-of-Data or End-of-Volume on Input

The system's EOVR routine handles both end-of-data-set and end-of-volume conditions on input. These conditions occur when:

- A tapemark is read.
- An FEOV (force-end-of-volume) macro is executed by the processing program.

After encountering a tapemark, the system checks the first 4 bytes of the first trailer label for the EOVR1 or EOF1 identifier. If neither identifier is found, processing is abnormally terminated. When the FEOV macro is executed, the identifier of the first trailer label is not checked. The trailer labels on the current volume are not processed but the volume labels on the next volume are processed.

For an end-of-data condition, the first trailer label is processed, unless deferred user input trailer label processing is specified in the DCB exit list. For an end-of-volume condition, the first trailer label on the current volume is processed, and then the volume label and header labels on the next volume are processed.

Except when it is used as a header label for a RDBACK operation, the system ignores the second trailer label (EOVR2 or EOF2) of an input data set.

If UTLs are present on input, the system can make them available to your program. To make them available, SUL must be coded on the DD statement and the address of an input user trailer label routine must be specified in the DCB exit list.

Verifying Block Count in Label When Reading

To verify that all records on the input data set on the current volume have been read, the system compares the block count shown in the first trailer label (EOV1 or EOF1) against the block count that was accumulated in the data control block. For reading backward, the system compares the 0 block count shown in the HDR1 label against the block count in the data control block.

If the block count in the label does not equal the block count in the data control block, the EOV routine gives control to the appropriate entry in the user's DCB exit list. This entry in the exit list is identified as X'0B'. The EOV routine passes the following information to the exit routine:

Register	Contents
0	The block count shown in the label
1	The address of the data control block.

After your exit routine analyzes the discrepancy (and possibly prints a message), your exit routine must return to the EOV routine with one of the following return codes in register 15:

Return Code	Meaning
0	Abnormally terminate with completion code of X'237'
4	Continue processing

For more information, see *DFSMS/MVS Using Data Sets* and "Verifying Block Count" on page 11.

If the appropriate user exit entry is not provided in the DCB exit list, or a return code of 0 is received, then a block count discrepancy causes the system to call the optional DCB ABEND exit and issue ABEND 237. When the FEOV macro is executed, the block count is not verified. For more information, see *DFSMS/MVS Using Data Sets*.

If the data set was created with an EXCP DCB with no device-dependent section, then the block count is written as zero and is not verified.

The EOV2/EOF2 Label

The system ignores the EOV2 or EOF2 trailer label of an input data set, except when it is used as a header label for a read backward operation.

User Trailer Labels

If user trailer labels (UTL1-UTL8) are present on input, the system can make them available to your program. To make them available, SUL must be coded on the DD statement and the address of an input user trailer label routine must be specified in the EXLST.

Determining Volume Switch

For a multivolume input data set, you must specify the serial numbers of all the volumes to be processed. The serial numbers are specified either directly in the DD statement or indirectly through the catalog procedure. Specify the serial numbers in forward sequence, regardless of whether the tapes are to be read forward or backward.

- For noncataloged data sets, you specify the volume serial numbers in the VOLUME parameter of the DD statement. The system processes the group of volumes in whatever order you specify and processes only the volumes you specify.
- For cataloged data sets, the group of volumes must be processed in sequential order. Processing can begin at any volume of the group by specifying a volume sequence number in the VOLUME parameter of the DD statement.

For input, the label identifier of the trailer labels determines whether the system continues processing the data set. When the system finds an EOV label, it performs volume switching. When the system finds an EOF label, it passes control to the user's end-of-data routine, with these exceptions:

- If the DD statement specifies OPTCD=B, and if an additional volume is available, then the system performs volume switching. If OPTCD=B, no volumes are available for switching, and no data set concatenated after the current one, then the system passes control to the user's end-of-data routine.
- If another data set is concatenated after this one, the system processes it. If the user has not set on the unlike-attributes bit (the DCBOFPPC bit), then the system treats the concatenated data set as if it began another volume of the current data set. If the user defined an EOV exit, the system calls it (not a DCB open exit). If the unlike-attributes bit is on, then the system closes the current data set and opens the concatenated data set. If the user defined a DCB open exit, the system calls it (not an EOV exit). For more information on using concatenated data sets, see *DFSMS/MVS Using Data Sets*.

To determine whether additional volumes are required, the system maintains a volume sequence number in the DEB in storage.

- For read forward operations, the volume sequence number in the DEB is increased as each volume is processed. This count is compared to the total number of volumes requested, as shown in the JFCB.
- For read backward operations, the volume sequence number in the DEB is initialized to the total number of volumes requested, as shown in the JFCB. The DEB count is decreased as each volume is processed until the count reaches 0.

If another volume is not required (end-of-data-set condition), control is given to the user's end-of-data routine that is specified in the data control block. Subsequently, the processing program or the operating system closes the data set.

- The user's end-of-data routine is not entered until the last specified volume or the last concatenated data set is processed.
- If an input data set is closed before the end of the data is reached, the user's end-of-data routine is not entered.

If another volume is required (end-of-volume condition), the system obtains the next volume serial number from the JFCB and performs volume switching. If the new volume is not already mounted, the EOV routine issues a mount message to the operator.

When multiple tape units are being used, the EOV routine also checks to see if a next-plus-one volume is specified, and if the volume just completed can be rewound and unloaded. If so, the EOV routine issues a message directing the oper-

ator to mount the next-plus-one volume on the tape unit just used. This is a pre-mounting aid; the next-plus-one volume label is not verified at this time. This lookahead or parallel mounting will result in an IEC501E mount message.

Note: The IEC501E message has a type 2 descriptor code which will cause the message to be retained on the screen.

If you do not want message retention, turn off the message retention attribute. You can accomplish this by either:

- Specifying RETAIN(NO) for the message id in an MPFLSTxx record
- Using a WTO message exit (an MPF specific exit or IEAVMXIT) to alter the message descriptor code or to request “do not retain this message.”

For tape mounts, the IEC501E message will remain on the screen until the mount is satisfied or until EOV has occurred on the previous volume. EOV will cause a delete operator message (DOM) to be initiated against the IEC501E message and will issue the IEC501A message.

Checking the Next Volume

When volume switching is performed, the EOV routine checks the volume and header labels on the new volume.

The VOL1 label is checked as if it were the first volume of the group; that is, the volume serial number is verified to ensure that the correct volume is mounted.

RACF access authorization is required for system-wide RACF tape protection. READ authorization is required if the volume is opened for INPUT or RDBACK. UPDATE authorization is required if the volume is opened for INOUT, OUTIN, or OUTINX. If a multivolume data set is RACF protected, then each volume data set must have the same profile for INOUT, OUTIN, or OUTINX (not INPUT or RDBACK) access.

No authorization is required if the new volume is not RACF protected and the user is authorized.

A specific volume request fails if the new volume is RACF protected and the user is unauthorized.

For a nonspecific volume request, the operator mounts a volume.

If the user is READ authorized but not UPDATE authorized, and the data set is open for INPUT or RDBACK, the system verifies that the volume is set to prevent writing. If the volume is not protected, the operator is notified to prevent the tape volume from writing by removing the write-enable ring from the reel or changing the write-protect tab on the cartridge. “Protecting Data” on page 43 describes when operator intervention is necessary, and when other mechanisms of protection eliminate the need for it.

The method of locating and checking the HDR1 label varies according to the situation. The processing depends on whether the data set is a continuation of a multivolume data set or is a concatenated data set. In either case, the user exit for security verification may be taken, as described in “Open/EOV User Exit for Security Verification” on page 47.

- *Multivolume data set:* The data set sequence number is irrelevant for the second and subsequent volumes of a multivolume data set. The EOVR routine assumes that the data set continues at the beginning of the new volume. Therefore, the first header label group on the tape is checked. The HDR1 label is checked in the same manner as when the data set was opened on the first volume. Processing is abnormally terminated if the data set name is not the same, or the password is not verified for each volume.

Note: For volumes residing in an IBM automated tape library, or manual tape library, password protection processing of the data tape set is bypassed. These tapes can be protected with RACF.

- *Concatenated data sets:* The EOVR routine handles concatenated data sets with like characteristics. Such data sets are not necessarily the first on the volume, so the EOVR routine positions the tape according to the specified data set sequence number. This positioning is the same as for opening a data set. The HDR1 label is checked in the same manner as when the first data set was opened, including verification of the password if protection is indicated.

Note: For volumes residing in an IBM automated tape library, or manual tape library, password protection processing of the data tape set is bypassed. These tapes can be protected with RACF.

The HDR2 label on each volume after the first volume within a multivolume data set is not processed. The data set characteristics that were established when the data set was opened apply to all subsequent volumes handled by the EOVR routine.

The data set's block count is not accumulated from volume to volume. It is initialized and verified separately for each volume.

Closing an Input Data Set

The close routine does not process trailer labels on an input data set. Usually, the trailer labels are processed by the EOVR routine before the data set is closed, unless deferred user input trailer label processing was specified in DCB exit list. If deferred user input trailer label processing was specified, the processing otherwise performed for an input end-of-data condition is performed when an input data set is closed.

If an input data set is closed before it reaches the end-of-data or the end-of-volume, or if the FEOVR macro is executed, processing of trailer labels is omitted.

Opening an Output Data Set

If IBM standard labels are specified, the first existing record on the output tape must be an IBM standard volume label (VOL1). At the time the data set is opened, the system checks the first existing record on the tape to determine whether the record is 80 bytes long and contains the identifier VOL1 in the first 4 bytes. The various error conditions that can occur during verification of the first record are explained in "IBM Standard Volume Label Verification and Editor Routines" on page 43.

If the system-wide RACF tape protection option has been specified and the DD statement has specified PROTECT=YES, and the DD statement has not been previously opened for output processing, open ensures the PROTECT=YES specifica-

tion is valid. Both the volume sequence number and the file sequence number must be set to 1, and a private volume must be requested. The protection indicator in the JFCB is reset so that subsequent OPENs of that DD statement for output processing do not attempt validity checking (of the PROTECT=YES specification) and defining the volume to RACF.

Attention: Doing multiple opens and closes without writing any user data in the end-of-tape reflective marker area can result in the writing of header and trailer labels past the reflective marker. Access methods detect the reflective markers; however, since the creation of empty data sets does not involve access methods, the end-of-tape marker is not detected. This can cause the reel to run out of tape.

Volume Serial Number

You are not required to specify volume serial numbers for output tapes, unless you are extending an existing tape data set. Serial numbers are required when the processing method is EXTEND, OUTINX, OUTPUT DISP=MOD. If no number is specified, the mount message directs the operator to mount a scratch tape. The system obtains the volume serial number from the VOL1 label and records it in the JFCB and the UCB.

A user exit identifies a specific tape volume in place of a nonspecific volume. The exit is invoked when open or EOVS issues a mount request for a volume with no volume serial number. For more information on the Open/EOVS nonspecific tape volume mount exit, see *DFSMS/MVS Using Data Sets*.

If you choose to specify the volume serial number, the system compares it with the volume serial number shown in the VOL1 label. When the serial numbers match, the system resets a mount switch in the unit control block to indicate that the mounted volume is verified. The mount switch is initially set when the mount message is issued to the operator. When the volume serial numbers do not match, the system may give the operator the option of having the label rewritten with the serial number of the volume requested. This occurs if the tape is not password protected, is not date protected, and is not a checkpoint/restart volume. DFSMSrmm prevents an incorrect tape from being used and the label being rewritten. Otherwise, the system rejects the tape and issues another mount message. Volumes are requested for mounting in the specified order .

If the system-wide RACF tape protection option has been specified, RACF authorization at the UPDATE level is checked. If the tape volume is not defined to RACF, processing continues. If the tape volume is defined RACF, and the user is UPDATE authorized and PROTECT=YES has not been specified, processing also continues.

If the tape volume is defined to RACF, the user is UPDATE authorized, and PROTECT=YES has been specified, processing will be abnormally terminated. If the tape volume is defined to RACF and the user is not UPDATE authorized, processing will also be abnormally terminated. When processing is abnormally terminated, the volume will be rejected if a nonspecific request is made, or the program will be abnormally terminated if a specific request is made.

Note: Beginning with RACF 1.7, if you specify PROTECT=YES in a DD statement for a volume or data set that you previously protected in this manner, you do not abnormally terminate due to this latter PROTECT=YES specification.

Positioning the Volume to the Data Set

When the volume is mounted and verified, the system positions the tape to receive the new data set. Usually the new data set is the first and only data set on the tape, so the tape remains positioned immediately following the VOL1 label.

To create a data set that follows another data set already stored on the tape, you specify a data set sequence number in the LABEL parameter of the DD statement.

- The sequence number can be from 1 to 9999, with 1 representing the first data set on the volume. If the volume ends with EOVS labels before the specified sequence number, the open routine switches to the next volume. If you specify a sequence number that is greater than the number of data sets existing on the volume, plus one, your task is abnormally terminated.
- If you do not specify a sequence number, or specify zero, the system assumes that the data set is to be written as the first on the volume.

To position the tape, the system maintains a logical data set sequence number in the UCB. The method of positioning is the same as that previously explained for opening an input data set.

Only one data set on a tape volume can be open at any given time. Processing is abnormally terminated if you attempt to open another data set on the same volume.

When the tape is positioned to receive the new data set, the system expects to find either an existing HDR1 label or a tapemark. If neither is present, the system assumes that other data is recorded where the HDR1 label should be, and processing is therefore abnormally terminated. If the last data set on a tape has EOVS labels, another data set cannot be written to follow it.

If a tapemark is found, it indicates that a HDR1 label does not exist at the position at which the new data set is to be written. The system bypasses all further label verification and accepts the tape for output. The conditions under which the system finds a tapemark instead of a HDR1 label are:

- When a tapemark immediately follows the VOL1 label. This can occur when the tape is initialized by means other than the EDGINERS or IEHINITT utility programs (EDGINERS and IEHINITT write a dummy HDR1 label following the VOL1 label). The tapemark is overwritten by the new HDR1 label.
- When the new data set is to be written after the last existing data set on the volume (for multiple data set organizations). In this case, the system encounters the second tapemark following the existing EOF trailer label group. The tapemark is overwritten by the new HDR1 label.

If the system finds an existing HDR1 label, it checks the label to determine whether the existing data set may be overlaid.

High Speed Search on an IBM Tape Cartridge

Positioning to the End of a Data Set: When you want to extend a data set written on a tape cartridge by an IBM magnetic tape subsystem using DISP=MOD, OUTINX, or EXTEND, you can use the *high speed search* function. This function quickly positions the volume to the end of the requested data set. The tape moves at approximately twice the normal transport speed. After the tape is positioned, the open module processes the trailer labels of the data set to be extended.

To invoke the high speed search function when extending a data set, you use the high speed search interface. Do the following:

1. Obtain the JFCB for the requested data set by using the read JFCB—RDJFCB—macro.
2. Set the “fast positioning” indicator in the JFCB (JFCPOSID in JFCBFLG3).
3. Set the block identifier in the JFCB (JFCRBIDO=blk-id).
4. Execute OPEN TYPE=J with the modified JFCB.

The block identifier for high speed positioning is for the tapemark immediately following the last block of user data. You can obtain the block identifier when the data set is created by issuing the NOTE macro with the ABS parameter before issuing CLOSE. This can be done only with BSAM or EXCP, not with QSAM.

Positioning to the Beginning of a Data Set: A similar method is used when you want to use the high speed search function to quickly position the volume to the beginning of the data set. The difference is that the block identifier is for the first standard header label of the requested data set when positioning to the beginning of the data set.

For more information about the high speed header label search function, see *3480 Magnetic Tape Subsystem: User's Reference*.

If the *fast positioning* indicator is set to on in a JFCB used by an open routine that is not TYPE=J, the fast positioning indicator will be reset.

When fast positioning is indicated, but a block identifier is not specified in JFCRBIDO, OPEN TYPE=J positions the tape normally and inserts a block identifier in JFCRBIDO. The block identifier is either for the first header label when opening to the beginning of a data set or for the tapemark immediately following the last block of user data when opening to extend a data set. You can use this technique with BSAM, QSAM, and EXCP.

OPEN does not update your copy of the JFCB. To retrieve the new value in the system's copy of the JFCB, issue RDJFCB after OPEN.

Once you have turned on the fast positioning indicator in a JFCB for an OPEN with TYPE=J in a job step, make sure the fast positioning indicator and the block identifier in JFCRBIDO reflect your intentions before any subsequent OPEN with TYPE=J for the same data set.

After OPEN with TYPE=J uses JFCRBIDO for a high speed search, it clears JFCRBIDO in the system copy of the JFCB to prevent misinterpretation in a subsequent OPEN.

Open/EOV User Exit for Security Verification

For APF-authorized programs for which the program property “bypass password (and RACF) checking” is active, a user exit is provided for verifying that a tape selected by open or EOV should be used, and whether certain security checks may be bypassed. For more information on the “Open/EOV volume security and verification” exit, see *DFSMS/MVS Installation Exits*.

Expiration Date on Existing Label

The existing HDR1 label is inspected for the expiration date. If it is an SMS-managed scratch volume, then the system ignores the expiration date in the label. Otherwise, if the expiration date has not been reached, the system asks the operator to confirm use of the tape or to mount another tape. DFSMSrmm provides facilities that automate the handling of expiration date protection. See *DFSMS/MVS DFSMSrmm Implementation and Customization Guide* for more information.

If other data sets exist on the same volume, the system checks only the one expiration date and assumes that all following data sets expire on the same date.

Password Protection and Data Set Name on Existing Label

After checking the expiration date, the system inspects the security indicator in the existing HDR1 label. This indicator shows whether the existing data set is protected against unauthorized use.

If no protection is indicated, or if protection is indicated but the volume resides in an IBM automated tape library, or a manual tape library, the tape is accepted for output. There is no password protection for IBM system-managed tape libraries.

If protection is indicated, the system compares the data set name shown in the existing HDR1 label to the name specified by the user in the DD statement. If the names are not the same, processing is abnormally terminated unless the data set is the first one on the first or only volume. In this case, even if you specify a specific volume, the operator will be requested to remove the tape and mount a new scratch tape. If a security-protected data set is deleted, the data set security byte in the HDR1 label must be set to 0 before the volume can be written on again. This can be done by using the EDGINERS or IEHINITT utility programs, or a user program to relabel the volume.

Two additional restraints are placed on creation of password-protected data sets:

- If you want to create a password-protected data set following an existing password-protected data set, you must supply the password of the existing data set. The security indicator must be the same in both the existing and the new data set. This consistency test is made even if the volume is RACF defined, or resides in an IBM automated tape library or manual tape library.
- When creating a multivolume, password-protected data set, the second and successive volumes will also be verified. Verification consists of ensuring that the data set name in the JFCB is the same as the data set name in the password record and that the protection-mode indicator allows writing to the data set.

If the data set name is correct, and if the tape volume has not been found to be RACF defined, the system requests the operator or TSO terminal user to key in the required password. The password is verified in a user-established password data set. This password data set contains the data set name, the password, and a protection-mode indicator. The protection-mode indicator is set to permit either read/write or read-only operations. The read/write mode is necessary for output data sets. Processing is terminated if:

- The operator or TSO terminal user, in two attempts, does not supply the correct password.

- The password record for the data set to be opened does not exist in the password data set.
- The read-only protection mode is specified.

DFSMS/MVS DFSMSdfp Advanced Services describes data set protection in detail and contains the information you need to create and maintain the password data set.

Note: Verification of existing labels is considered complete after checking the HDR1 label. Any labels, data, data sets, or tapemarks following the HDR1 label are irrelevant and may be overlaid by the new output.

Writing Data Set Header Labels

When the tape is accepted by the system for output, the system creates the header labels (HDR1 and HDR2) for the new data set. These labels are created from information in the updated JFCB and other system control blocks.

The source of information for the label fields is explained in the description of the label formats. The process of updating the JFCB is explained in Figure 2 on page 4.

The security indicator is set in the HDR1 label even if the volume is RACF defined, unless the volume resides in an IBM automated tape library in which case there is no password protection.

If no user header labels are to be written, the system writes a tapemark after the HDR2 label. The tape is then ready to receive the new data set.

Writing User Header Labels

When SUL is coded on the DD statement and the address of an output user header label routine is specified in the DCB exit list, the system can write as many as eight user header labels (UHL1 to UHL8).

Permanent I/O Error

If a permanent I/O error occurs during label processing, and the data set is the first one on the first or only volume, the operator will be requested to remove the tape and mount a scratch tape. The job will be abnormally terminated if the data set is not the first one on the volume or the volume is not the first volume of a multi-volume data set.

End-of-Volume on Output

The system's EOV routine automatically switches volumes when one of the following EOV conditions occur:

- The reflective strip on a reel is encountered
- The logical end of a tape cartridge is reached
- The FEOV macro is executed.

This volume switching includes:

1. Checking whether the expected number of blocks were read or written (see "Verifying Block Count" on page 11)
2. Writing trailer labels on the current volume

3. Checking existing volume and header labels on the new volume
4. Writing a volume label on the new volume (in the case of a density conflict or version conflict)
5. Writing header labels on the new volume.

When multiple tape units are used, the EOVRoutine also checks to see if a next-plus-one volume is needed, and if the volume just written can be rewound and unloaded. If so, the EOVRoutine issues a message directing the operator to mount the next-plus-one volume on the tape unit just used. This is a premounting aid; the next-plus-one volume label is not verified at this time. This lookahead or parallel mounting results in an IEC501E mount message.

Note: The IEC501E message has a type 2 descriptor code that causes the message to be retained on the screen.

If you do not want message retention, turn off the message retention attribute. You can accomplish this by either:

- Specifying RETAIN(NO) for the message id in an MPFLSTxx record
- Using a WTO message exit (an MPF specific exit or IEAVMXIT) to alter the message descriptor code or to request “do not retain this message.”

For tape mounts, the IEC501E message remains on the screen until the mount is satisfied or until EOVRoutine has occurred on the previous volume. EOVRoutine causes a DOM to be initiated against the IEC501E message and issues the IEC501A message.

Writing Data Set Trailer Labels

Trailer labels are always written at an end-of-volume condition on output tapes. These labels are identified as EOVR1 and EOVR2 (as opposed to EOF for end of data). These labels are created in the same manner and with the same content as the data set header labels, except for the label identifiers and the block count.

At end of volume, one tapemark is written following the data set trailer labels. If user trailer labels are to be written, the tapemark follows the user labels.

Writing User Trailer Labels

When SUL is coded on the DD statement, and the address of an output user trailer label routine is specified in the DCB exit list, the system can write as many as eight user trailer labels (UTL1 to UTL8).

Writing Labels on New Volume

The EOVRoutine handles label processing on the new volume (checking existing labels and writing new labels). The processing is the same as the open routine's handling of the first volume. The user exit for security verification may be taken, as described in “Open/EOVRoutine User Exit for Security Verification” on page 56.

When creating a multivolume data set, the data set sequence number is irrelevant for the second and subsequent volumes. The EOVRoutine assumes that the data set continues at the beginning of the new volume.

RACF Processing on the New Volume

If the system-wide RACF tape protection option has been specified, RACF authorization checking will occur for the new volume. If the previous volume is RACF protected, the new volume must either be not defined to RACF (in which case EOV will define it to RACF as part of the previous volume's volume set profile), or the new volume must be defined as part of the same volume set profile as the previous volume. If the previous volume is not RACF protected, the new volume must not be RACF protected. If the above conditions are not met, the new volume is rejected when a nonspecific request is made, or the program is abnormally terminated when a specific request is made.

Special End-of-Volume Conditions

The following conditions signal the end of the volume:

- The drive signals to the system that it is nearing the end of the tape.
 - On a reel, the system detects the end-of-tape reflective marker.
 - On a cartridge, internal mechanisms sense the end of the tape.
- An FEOV command is issued.

The EOV routine writes the trailer labels as described above. If the drive signal occurs while the system is writing the trailer labels, the EOV or the close routine continues to write the trailer labels. In both cases, the data set can be read or overwritten normally.

If the system is writing header labels, and the drive signals that it is nearing the end of the tape, the system:

- Writes the new header label group
- Allows the user to write one record
- Writes the new trailer label group
- Performs volume switching.

Closing an Output Data Set

The close routine handles end-of-data-set processing on output tapes. When a write operation is the last operation that occurs before closing a data set (for OUTPUT, EXTEND, OUTIN, OUTINX, or INOUT) or when no output is written before closing (for OUTPUT, OUTIN, or OUTINX), the close routine creates data set trailer labels.

Writing Data Set Trailer Labels

The close routine writes the data set trailer labels with the identifiers EOF1 and EOF2. Except for the label identifiers and the block count, these labels are created in the same manner and with the same content as the data set header labels.

The close routine writes two tapemarks following the trailer labels. If user labels are to be written, the tapemarks follow the user trailer labels. If another data set is added to the tape (multiple data set organization), its HDR1 label overlays the second tapemark.

Writing User Trailer Labels

When SUL is coded on the DD statement and the address of an output user trailer label routine is specified in the DCB exit list, the close routine can write as many as eight user trailer labels (UTL1-UTL8).

Restarting from a Checkpoint

When a job step is restarted from a checkpoint, the restart routine repositions tape volumes containing data sets that were open when the checkpoint was taken. Specifically, the restart routine:

- Restores applicable control blocks to the conditions that existed when the checkpoint was taken.
- Ensures that the first existing record on the tape is a standard volume label (VOL1), and verifies the volume serial number shown in the label. If the volume is a reel:
 - Uses the data set sequence number shown in the JFCB to position the tape at the interrecord gap preceding the first record of the required data set. The method of positioning is the same as explained in “Positioning the Volume to the Data Set” on page 45. The data set labels are not reprocessed.
 - Uses the block count shown in the DCB to reposition the tape at the proper record within the data set. This positioning is always performed in a forward direction. If the block count is 0 or a negative number, the tape remains positioned at the interrecord gap preceding the first record.
- If the volume is a cartridge then the block identifier saved in the checkpoint is used to position the tape at the proper record.

If a SYSOUT data set was open when the checkpoint was taken, the data set written into during restart differs from the data set used originally. The system writes job separators at the beginning of the SYSOUT data set used during restart.

Note: For the IBM Magstar 3590 Tape Drive, restart will issue a read previous command instead of a read backward command.

Chapter 3. ISO/ANSI Labels

This chapter describes the organization, format, and processing of labels designed according to the specifications of the following industry standards as understood and interpreted by IBM:

- ANSI X3.27-1978, level 4
- ISO 1001-1979, level 4
- FIPS 79.

and

- ANSI X3.27-1987, level 4
- ISO 1001-1986(E), level 4

The term *Version 4* is used when referring to ANSI X3.27–1987 level 4 and ISO 1001–1986(E) level 4 standards.

In this book, the term *Version 3* is used when referring to ANSI X3.27–1978, ISO 1001–1979 and FIPS 79 standards.

The term *Version 1* is used when referring to the previous ISO 1001-1967 and ANSI X3.27-1969 standards.

Only volumes with Version 3 or Version 4 labels will be created by the system. Volumes with Version 1 labels will be accepted for input, but Version 3 or Version 4 checking will not occur.

ISO/ANSI support executes on a processor with a magnetic tape device that supports the specifications in:

- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 32rpm (800rpi), ISO 1863*
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 8 and 32rpm (200 and 800rpi) NRZI, and 63rpm (1600rpi), Phase Encoded, ISO 1864*
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 63rpm (1600rpi), Phase Encoded, ISO 3788*
- *American National Standard Recorded Magnetic Tape for Information Interchange, 800cpi, NRZI, X3.22-1973*
- *American National Standard Recorded Magnetic Tape for Information Interchange, 1600cpi, PE, X3.39-1973*
- *American National Standard Recorded Magnetic Tape for Information Interchange, 6250cpi Group-Coding, X3.54-1976*

Both *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 8rpm (200rpi), ISO 1862*, and *American National Standard Recorded Magnetic Tape for Information Interchange, 200cpi, NRZI, X3.13-1973*, require a magnetic tape device not supported by MVS/ESA*. Tapes written according to those standards are not supported.

All data on a tape with Version 3 labels must be recorded in the 128 characters of basic ASCII. Also, all the data in a Version 4 tape label must be recorded in the 128 characters of basic ASCII. Do not use Version 3 or Version 4 volumes to contain raw binary, floating point, or packed decimal data. See “ASCII Conversion” on page 99 for more information. Unlabeled tapes recorded in ASCII can be processed, as explained in Chapter 5, “Unlabeled Tapes” on page 121.

Label Definition and Organization

ISO/ANSI labels are similar to IBM standard labels. The principal differences are:

- ISO/ANSI labels are written in ASCII characters. IBM standard labels are written in EBCDIC, except on 7-track tape. IBM standard labels on 7-track tape are written in BCDIC.
- IBM standard labels are 80 bytes long. ISO/ANSI labels have a length of 80 bytes or more.
- A maximum of 9 user volume labels can appear in the beginning-of-volume group. Version 4 labels can also have VOL2–VOL9 appearing after VOL1 label in the beginning-of-volume group.
- An unlimited number of ISO/ANSI user labels can be placed at the beginning and end of a file, and they need not be sequentially numbered or lettered.
- The formats of the ISO/ANSI labels VOL1, HDR2, EOF2, and EOVS are slightly different from the formats of the corresponding IBM labels.
- A maximum of 9 user EOF or EOVS labels can appear in the file section label group.

The labels must be recorded in the subset of ASCII characters allowed by ISO/ANSI standards.

For Version 3:

- Uppercase alphabetic

For Version 4:

- Uppercase alphabetic
- Numeric
- Space
- Special (specifically, ! " % & ' () * + , - . / : ; < = > ? _)

All fields in Version 3 or Version 4 system labels (VOL1, HDR1, HDR2, EOVS, EOVS, EOF1, and EOF2) will be treated as containing meaningful data. This means alphanumeric fields (except “reserved for operating system” fields) will be left-justified, with unused positions filled with space characters; “reserved for future standardization” fields will be filled with space characters; numeric fields will be right-justified, with unused positions filled with zeros. Date fields may have a leading space character.

The first 4 characters of an ISO/ANSI tape label always identify the type of label:

Label Identifier	Label Definition
VOL1–VOL9	Volume label set (optional; VOL2–VOL9 not produced by MVS)

UVL1-UVL9	User volume labels (optional; <i>not</i> produced by MVS)
HDR1	Data set header label 1
HDR2	Data set header label 2 (produced by MVS, but optional for input)
HDR3-HDR9	Optional (not produced by MVS)
UHLa	User header labels (optional; unlimited number permitted)
EOV1	End-of-volume trailer label 1 (produced by MVS, but optional for input)
EOV2	End-of-volume trailer label 2 (produced by MVS, but optional for input)
EOV3-EOV9	Optional (not produced by MVS)
EOF1	End of data set trailer label 1 (produced by MVS, but optional for input)
EOF2	End of data set trailer label 2 (produced by MVS, but optional for input)
EOF3-EOF9	Optional (not produced by MVS)
UTLa	User trailer labels (optional; unlimited number permitted)

Note: The fourth character of the user header and trailer labels may be any valid ASCII character as defined above.

Figure 14 on page 66 and Figure 15 on page 67 show the position of the labels with various tape volume organizations. User labels (UHL, UTL) and volume labels (VOL2–9) are optional.

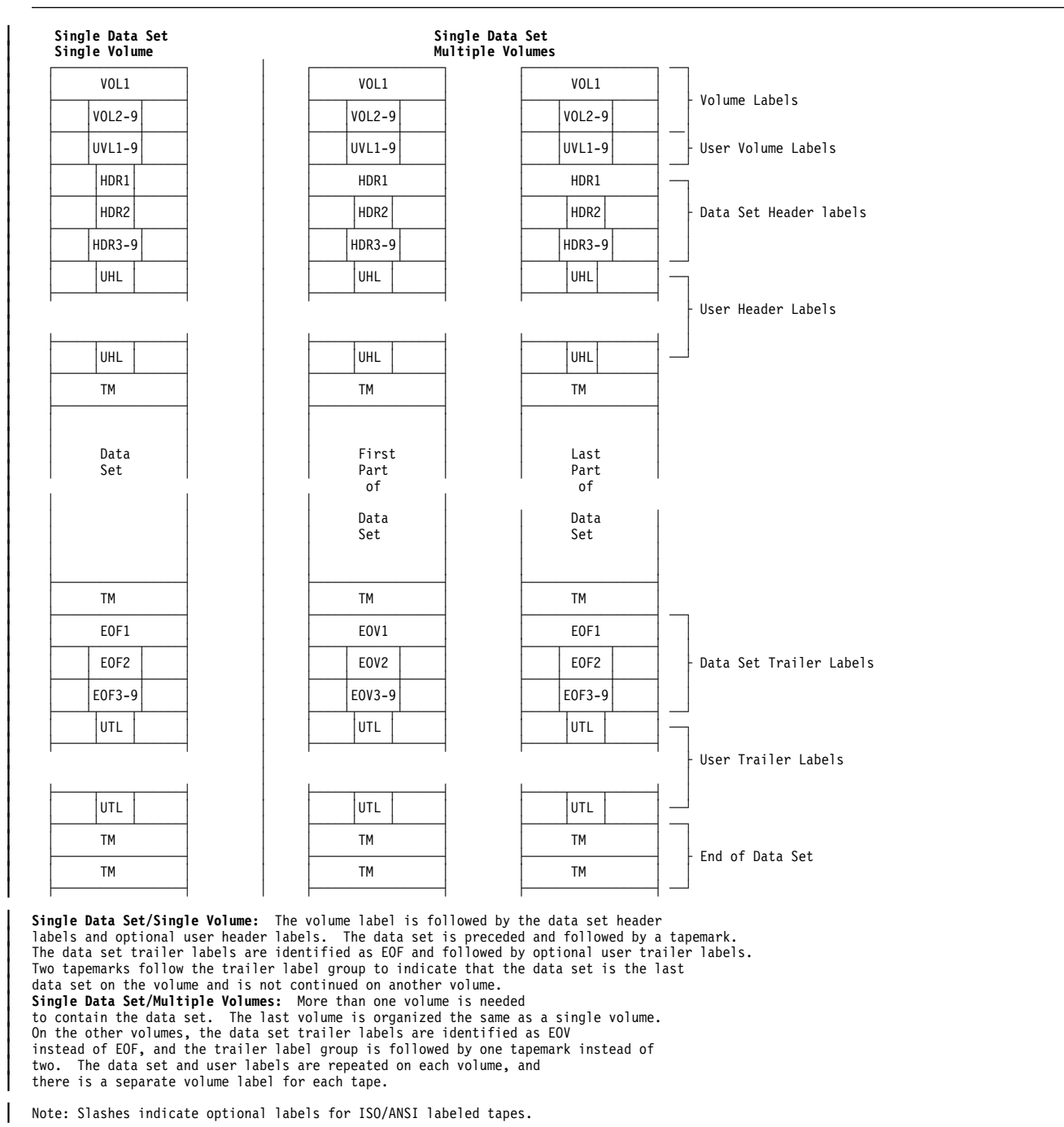


Figure 14. Volume Organization with ISO/ANSI Labels (Single Data Set)

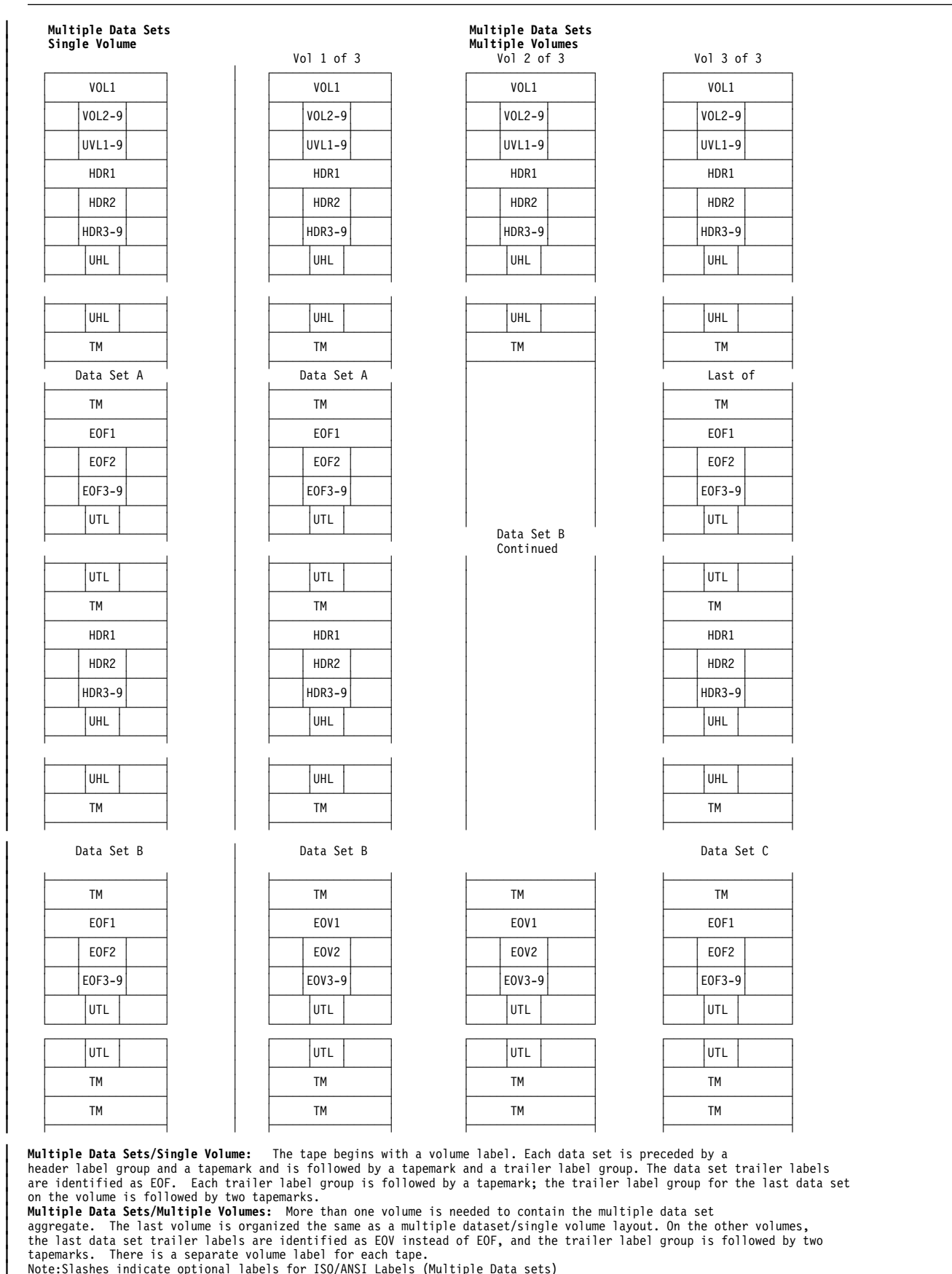


Figure 15. Volume Organizations with ISO/ANSI Labels (Multiple Data Sets)

Volume Label

The volume label (VOL1) is the first record on an ISO/ANSI labeled tape. It is at least 80 characters long. Although an ISO/ANSI label can exceed a length of 80 bytes, the excess is not within the scope of the standards and will be truncated by MVS/ESA. All Version 3 and Version 4 labels written by MVS/ESA routines will be 80 bytes in length, including those in which an original label greater than 80 bytes is rewritten (as in a density conflict).

The volume label identifies the volume and its owner and is used by the system to verify that the correct volume is mounted. It also protects the volume from unauthorized use.

Volume labels are created by the EDGINERS or IEHINITT utility programs, or by input from the operator during the Open or EOVS routine when:

- A label conflict is detected. A label conflict occurs when the type of label requested differs from the type of label read at load point from the mounted volume.
- The first block of a volume cannot be read (except a RACF-protected volume is rejected if failure to read was because the tape format, such as a nonsupported density, was not recognizable by the control unit).

For additional information about EDGINERS and the IEHINITT utility programs, see *DFSMS/MVS DFSMSrmm Implementation and Customization Guide*.

Volume labels (VOL2-VOL9) are allowed but not created or processed by MVS. They are checked only for valid label identification and correct sequencing.

User volume labels (UVL1-UVL9) are allowed but not created or processed by MVS. They are checked only for valid label identification and correct sequencing.

Data Set Header Labels

ISO/ANSI header labels consist of a data set label 1 (HDR1) and, in some cases, a data set label 2 (HDR2). The HDR1 label is created by the system routines each time a data set is recorded on tape. A HDR1 label identifies and describes the data set and protects it from unauthorized use.

The HDR1 label is 80 characters long. Any characters after position 80 are not used for label checking or verification.

A HDR2 label is optional for ISO/ANSI labeled tapes. When present, it immediately follows the HDR1 label. The MVS operating system produces a HDR2 label for output tapes; such a label on an input tape is treated similarly to an IBM HDR2 label. If the HDR2 label is produced by another system, the “reserved for operating system” fields are not processed.

Labels identified as HDR3-HDR9 are allowed but only checked for valid label identification and correct sequencing (and only if they are among the header labels of a requested data set). MVS does not create optional data set header labels beyond the second label.

User Header Labels

User header labels (UHLs) are optional, and there is no limit to the number that can be associated with a data set. They must contain only valid ASCII characters, but need not be sequentially numbered or lettered. These labels contain user-specified data that can be made available to your program for processing.

If you want the system to write user header labels or to make user header labels available to your program, you must specify AUL on the DD statement and the address of a user header label routine in the DCB exit list. (The exit list can address several user header label routines, that is, routines that process input user header labels and create output user header labels.) The DCB exit list is described in *DFSMS/MVS Using Data Sets*.

Data Set Trailer Labels

The data set trailer label 1 (EOV1, EOF1) is required for ISO/ANSI labeled tapes. The standard EOV1 and EOF1 labels duplicate the standard HDR1 label so that the tape can be read backward. The EOV1 and EOF1 labels are identical to a HDR1 label except that:

- The identifier is EOV or EOF instead of HDR.
- A block count is recorded in the first EOV1 or EOF1 label and is used on input to verify that all blocks of the data set are processed. The block count field in the HDR1 label contains zeros (which is what the EOF1/EOV1 block count is reduced to when the data set is read backward).

A data set can be extended by a job different from the one that started writing it. You might do that with DISP=MOD on the DD statement or with the OPEN macro options EXTEND or INOUT. On a Version 3 tape these are allowed only if the installation's label validation exit allows it. Extending a data set might result in fields such as these differing between headers and trailer labels: job name, step name, creation date, expiration date.

Eighty-character Version 3 or Version 4 EOV1 and EOF1 labels are created by the system when the data set is recorded on tape. EOV1 and EOF1 labels created by non-MVS systems must be up to 80 characters long; any characters after position 80 are not used for checking or verification.

Although they are optional for Version 3 labeled tapes, the EOV2 and EOF2 labels are produced by the operating system. These labels are treated similarly to IBM EOV2 and EOF2 labels. If the EOV2 or EOF2 labels are produced by another system, then the "reserved for operating system" fields are not processed.

Labels identified as EOV3-EOV9 or EOF3-EOF9 are allowed but checked for valid label identification and correct sequencing if they are among the trailer labels of a requested data set. The system does not create optional trailer labels beyond the second label.

User Trailer Labels

User trailer labels (UTLa) are optional, and there is no limit to the number that can be associated with a data set. They must contain only certain ASCII characters, but need not be sequentially numbered or lettered. These labels contain user-specified data that can be made available to your program for processing.

If you want the system to write user trailer labels or to make user trailer labels available to your program, you must specify AUL on the LABEL parameter of the DD statement and the address of a user trailer label routine in the DCB exit list. The exit list can address several user trailer label routines, that is, routines that process input user trailer labels and create output user trailer labels. The DCB exit list (EXLST) is described in *DFSMS/MVS Using Data Sets*.

System-Managed Mountable Tape External Labels

An external label is required for every tape cartridge in the tape library. This label contains both machine-readable and operator-readable identification characters that are used by the Tape Library Dataserver's vision system to identify the volume serial number of the cartridge. In a manual tape library, a machine readable external label is not required, but is advisable for migration purposes.

Each label can have a maximum of six uppercase characters for identification purposes:

- Uppercase A-Z
- Numerics 0-9
- Trailing blank space (non-service volumes only)

The external label must be applied within the indentation on the side of the cartridge; however, a larger information label, supplied by the customer, can be used on the top of the cartridge.

Tapemarks

Each data set and each data set label group to be processed by data management must be followed by a tapemark.

- There is no tapemark between the volume label and the first header label group on the volume.
- The tapemark that marks the end of the header label group also indicates the beginning of the data set to be processed.
- The tapemark that follows the data set also indicates the beginning of the trailer label group.
- A tapemark marks the end of the trailer label group. A second tapemark follows the trailer label group of the last data set on the volume, provided that the data set does not continue on another volume.

When the operating system is used to create a data set with IBM ISO/ANSI labels, it writes the necessary tapemarks.

ISO/ANSI Labels Format

The ISO/ANSI standard volume label is 80 bytes or longer and is always used to identify the tape volume and its owner. It is always the first record on ISO/ANSI standard labelled tape. It is recorded in ASCII.

Format of the ISO/ANSI Volume Label (VOL1)

Figure 16 on page 72 shows the format of the ISO/ANSI volume label. The shaded areas represent fields that are recorded in the label, but not used or verified during processing. The contents and processing of each field of the label are described, as are differences between the ISO/ANSI volume label and the IBM volume label.

Position	(Bytes)	Field Number and Name
3	(3)	1 Label Identifier ("VOL")
4	(1)	2 Label Number ("1")
5	(6)	3 Volume Identifier
10	(1)*	4 Accessibility
11	(13)*	5 Reserved for Future Standardization
12		
24		
25	(13)*	6 Implementation Identifier
37		
38	(14)*	7 Owner Identifier
51		
52	(28)*	8 Reserved for Future Standardization
79		
80	(1)*	9 Label Standard Level

* ISO/ANSI field differs from corresponding IBM field.

Figure 16. Format of the ISO/ANSI Volume Label

1—Label Identifier (3 bytes)

- *Contents:* The characters VOL identify this label as a volume label.
- *Processing:* This field is read to verify that a standard labeled tape is mounted, and that this label is a volume label.

The labels are created by the EDGINERS or IEHINITT utility programs.

In the following situations, the Open or EOVS routines create a Version 3 or Version 4 volume label based on specifications in your DD statement:

- If you request a Version 3 or Version 4 output volume (AL, AUL) without specifying a volume serial number, and an NL, NSL, or SL labeled volume is mounted to satisfy your request
- If you request a Version 3 or Version 4 output volume (AL, AUL) without specifying a volume serial number, and a volume that can be overwritten and recorded in the wrong density is mounted to satisfy your request (only if output is for the first data set on the volume).

For ISO/ANSI labeled tape volume requests, if OPEN or EOVS enters label editor routines, the VOL1 label can be written or rewritten. See “ISO/ANSI Volume Label Verification and Editor Routines” on page 94 for more information.

2—Label Number (1 byte)

- *Contents:* The relative position of this label within a set of labels of the same type; it is always 1 for the Version 3 or Version 4 volume label.
- *Processing:* Verified in conjunction with Field 1 to identify this label as VOL1.

3—Volume Identifier (6 bytes)

- *Contents:* A unique identification code, known in the system as the volume serial number, assigned through JCL, DFSMSrmm, or IEHINITT to the volume when it enters the system, or assigned by the operator when Open or EOVS routines label the volume. This code might also appear on the external surface of the volume for visual identification. The code is typically numeric (000000 to 999999), but can be from 1 to 6 characters long. If the code is less than 6 characters, it is left-justified and padded with blanks.

All national characters and some special characters in this field are rejected during open as invalid ASCII characters. For a list of the valid ASCII characters, see “Label Definition and Organization” on page 64.

- *Processing:* When the user specifies a volume serial number, it is in the JFCB. The system copies it to the UCB. When the system processes the labels, it compares the number in the UCB to the number in this field of the label to ensure that the correct volume is mounted.

For scratch output tapes, the volume serial number is obtained from this field of the label and recorded in both the JFCB and the UCB.

- *Difference from IBM Field:* The corresponding field in an IBM standard label is called “Volume Serial Number.”

4—Accessibility (1 byte)

- *Contents:* A valid Version 3 or Version 4 character indicates that the RACHECK installation exits are entered and receive accessibility parameters, or the volume access exit will be entered in order to accept or reject the volume. A space indicates that the volume is authorized for access, unless RACF rejects the volume. All other characters cause the volume to be rejected if the volume is not defined to RACF.
- *Processing:* If this field contains any invalid Version 3 or Version 4 characters, then the volume is rejected by the system, unless the volume has been defined to RACF.
- *Difference from IBM Field:* The corresponding field in an IBM standard label is reserved and currently unused.

5—Reserved for Future Standardization (13 bytes)

- *Contents:* Reserved for possible future use and recorded as blanks.
- *Processing:* Not used or verified, except to check for all blanks. The EDGINERS and IEHINITT utility programs write blanks in this field. (Blanks are translated to ASCII space characters on output.)

6—Implementation Identifier (13 bytes)

- *Contents:* Reserved for possible future use and recorded as blanks.
- *Processing:* Not used or verified, except to check for all blanks. The EDGINERS and IEHINITT utility programs write blanks in this field. (Blanks are translated to ASCII space characters on output.)

7—Owner Identifier (14 bytes)

- *Contents:* Indicates a specific customer, person, installation, department, and so forth, to which the volume belongs. Any code or name is acceptable.
- *Processing:* Not used or verified, except to check for valid Version 3 or Version 4 characters. The EDGINERS and IEHINITT utility programs write the text specified by the user, and the open and EOVRoutines write the text specified by the operator. If the code is less than 14 bytes long, it is left-justified and the remainder of the field is padded with blanks. (Blanks are translated to ASCII space characters on output.)
- *Difference from IBM Field:* The corresponding field on an IBM standard label is 10 bytes long.

8—Reserved for Future Standardization (28 bytes)

- *Contents:* Reserved for possible future use and recorded as blanks.
- *Processing:* Not used or verified, except to check for all blanks. The EDGINERS and IEHINITT utility programs write blanks in this field. (Blanks are translated to ASCII space characters on output.)

9—Label Standard Level (1 byte)

- *Contents:* 3 signifies that the tape is formatted according to Version 3 interchange standards. 4 signifies that the tape is formatted according to Version 4 interchange standards.
- *Processing:* The operating system always places 3 or 4 in this field on output. Version 1 tapes contain 1 in this field and are accepted for input processing. The system rejects any character other than 1, 3, or 4 in this field.
- *Difference from IBM Field:* This field is blank in IBM standard labels.

Format of ISO/ANSI Data Set Label 1 (HDR1/EOV1/EOF1)

Figure 17 on page 75 shows the format of HDR1, EOV1, and EOF1. The shaded areas represent fields that the operating system writes in the label, but are not used or verified during processing. The contents and processing of each field of the label are described, as are differences between the ISO/ANSI labels and the IBM labels.

Position	(Bytes)	Field Number and Name
1	(3)	1 Label Identifier
3		2 Label Number
4		
5	(17)*	3 File Identifier
21		
22		
	(6)*	4 File Set Identifier
27		
28	(4)*	5 File Section Number
31		
32	(4)*	6 File Sequence Number
35		
36	(4)*	7 Generation Number
39		
40	(2)*	8 Version Number
41		
42		
	(6)	9 Creation Date
47		
48		
	(6)	10 Expiration Date
53		
54	(1)*	11 Accessibility
55	(6)	12 Block Count
60		
61		
	(13)*	13 System Code
73		
74		
	(7)	14 Reserved for Future Standardization
80		

* ISO/ANSI field differs from corresponding IBM field.

Figure 17. Format of ISO/ANSI Header 1 and Trailer 1 labels

1—Label Identifier (3 bytes)

- *Contents:* Three characters that identify the label are as follows:

HDR Header label (at the beginning of a data set)

EOV Trailer label (at the end of a tape volume, when the data set continues on another volume)

EOF Trailer label (at the end of a data set).

- *Processing:* The system checks this field to verify that the record is an ISO/ANSI data set label.

For input data sets, the system checks the label identifier to determine whether data set processing should continue. When the system finds an EOVS label, it performs volume switching. When the system finds an EOF label, it passes control to the user's end-of-data routine or continues processing with a concatenated data set.

If the DD statement specifies OPTCD=B for an input data set, the trailer label identifier (EOVS or EOF) is not used to determine whether a volume switch is necessary. If more volumes are available, the system performs the switching. If no volumes are available, the system passes control to the user's end-of-data routine or continues processing with a concatenated data set.

When creating trailer labels, the EOVS routine writes EOVS in this field, and the close routine writes EOF.

2—Label Number (1 byte)

- *Contents:* The relative position of this label within a set of labels of the same type; it is always 1 for data set label 1.
- *Processing:* Verified and written in conjunction with Field 1 to identify this label as HDR1, EOVS1, or EOF1.

3—File Identifier (17 bytes)

- *Contents:* The rightmost 17 bytes of the data set name, not including the suffix .GxxxxVyy for generation data sets. Version 1 tapes will continue to be accepted for input with the suffix included as part of the file identifier. If the data set name is fewer than 17 bytes, it is left-justified and the remainder of this field is padded with ASCII space characters.

If the name contains embedded spaces or other special characters, you must enclose the name in apostrophes on the DD statement that requests this data set. *OS/390 MVS JCL User's Guide* lists the restrictions that apply to enclosing a data set name in apostrophes. The apostrophes do not appear in the data set identifier field.

- *Processing:* For input, this name is compared to the user-specified data set name found in the JFCB. This comparison ensures that the correct data set is being processed. For Version 3 only, the file identifier is also compared to the names of other data sets on the volume during open positioning to ensure against duplicate names. See "Processing the HDR1 Label" on page 106 for more information.

For output, the data set name in the existing label is verified in conjunction with password protection to determine whether the existing data set can be overwritten. If password protection is not specified, then the data set name is checked for valid Version 3 or Version 4 characters.

For input and output the first file data set identifier is compared with the last 17 characters of the first file data set name recorded in the DFSMSrmm control data set. If there is no match, the volume is rejected. This check ensures that the correct volume is mounted.

When creating labels for a new data set, the user-specified data set name is obtained from the JFCB and recorded in this field.

- *Difference from IBM Field:* The corresponding field in an IBM standard label is called "Data Set Identifier."

4—File Set Identifier (6 bytes)

- *Contents:* The volume serial number of the tape volume containing the data set. For multivolume data sets, this field contains the serial number of the first volume of the aggregate created at the same time.

If the volume serial number is specified in the JCL statements, all national characters and some special characters are rejected during the open routine as invalid Version 3 or Version 4 characters. See "Label Definition and Organization" on page 64 for a list of the valid Version 3 or Version 4 characters. If the code is fewer than 6 characters long, it must be left-justified.

- *Processing:* If this field has inconsistent values in the volumes in a volume set as described in Figure 6 on page 18 and Figure 7 on page 19, the system writes a warning message . The warning indicates that one of the volumes might be the wrong volume. The message number is IEC709I. When creating labels, the serial number is obtained from the UCB and recorded in this field.
- *Difference from IBM field:* The corresponding field on an IBM standard label is called "Data Set Serial Number."

5—File Section Number (4 bytes)

- *Contents:* A number (0001 to 9999) that indicates the order of the volume within the multivolume group created at the same time. This number is always 0001 for a single volume data set.
- *Processing:* If this field does not have a value one greater than the previous volume in a volume set as described in Figure 6 on page 18 and Figure 7 on page 19, the system writes a warning message IEC709I. This warns the volumes might be out of order. When creating labels, the open routine writes 0001 in this field; the EOVS and close routines write a number one higher than on the previous volume.
- *Difference from IBM Field:* The corresponding field on an IBM standard label is called "Volume Sequence Number."

6—File Sequence Number (4 bytes)

- *Contents:* A number (0001 to 9999) that indicates the relative position of the data set within a multiple data set group. This number is always 0001 for a single data set organization.
- *Processing:* This number in the first HDR1 label on the tape is referred to when the open routine positions the tape. If this number in the first HDR1 label and the requested data set sequence number in the JFCB are both greater than 1, then the logical data set sequence number in the UCB is set to the number in the label. Otherwise, the logical data set sequence number in the UCB is set to 1.

When creating labels, the open and close routines obtain the user-specified data set sequence number from the JFCB (0 is changed to 1). The EOVS routine propagates the number.

7—Generation Number (4 bytes)

- *Contents:* If the data set is part of a generation data group, this field contains a number from 0001 to 9999 indicating the absolute generation number (the first generation is recorded as 0001). If the data set is not part of a generation data group, this field contains 0001.
- *Processing:* A nonnumeric or all zero value is not valid, and can cause the label validation exit to be entered unless validation has been suppressed.

When creating labels, the system checks the JFCB to determine whether the data set is part of a generation data group. If so, the generation number is obtained from the last part of the data set name in the JFCB. Otherwise, this field is recorded as 0001.

8—Version Number (2 bytes)

- *Contents:* If the data set is part of a generation data group, this field contains a number from 00 to 99 indicating the version number of the generation (the first version is recorded as 00). If the data set is not part of a generation data group, this field contains ASCII zeros.
- *Processing:* The system always records this field as zeros. For a version level other than zero, you must specify the absolute generation and version numbers as part of the data set name when creating or retrieving a data set.

9—Creation Date (6 bytes)

- *Contents:* Date when allocation begins for creating the data set. The date is format cyydd, where:

c = century (blank=19; 0=20; 1=21; etc.)
yy = year (00-99)
ddd = day (001-366)

Note: The century code gives the first two digits of the year, not the actual century. For example, a blank which translates into 19 indicates a year in the 1900s, not in the nineteenth century.

- *Processing:* Not used or verified, except to check for proper format. When the system creates labels, the date is obtained from the JFCB. If a data set is allocated via JCL, then the creation date is the date when allocation begins for the job step responsible for creating the data set. If a data set is allocated dynamically, then the creation date is the date when allocation begins for the data set.

10—Expiration Date (6 bytes)

- *Contents:* Year and day of the year when the data set may be scratched or overwritten. The data is shown in the format cyydd, where:

c = century (blank=19; 0=20; 1=21; etc.)
yy = year (00-99)
ddd = day (001-366)

Note: The century code gives the first two digits of the year, not the actual century. For example, a blank which translates into 19 indicates a year in the 1900s, not in the nineteenth century.

- *Processing:* For input, not used or verified, except to check for proper format. For output, the expiration date in the existing label is compared to the current date. If the date in the label is later than the current date, then the operator receives a message and has the option of using the tape or mounting another.

If other data sets are on the same volume, then the system checks the expiration date of the immediately preceding data set. If the previous data set's expiration date is earlier than that of the output data set, then the label validation exit is entered (unless label validation has been suppressed). If any other data sets follow on the same volume, then they are considered expired on the same day. The expiration dates of scratch SMS-managed tape volumes, both automatic and manual tape libraries, are ignored. DFSMSrmm can override the expiration date or reject the volume without operator intervention.

If you enter 99365 or 99366, the system retains your data sets permanently. Do not use those dates as expiration dates. Use them as no-scratch dates only.

When creating labels, the system obtains the expiration date from the JFCB. If you did not specify a retention period or expiration date, then the expiration date is recorded as zeros, and the data set expires. DFSMSrmm uses the expiration date to update information its control data set. For data sets without an expiration date, DFSMSrmm assigns one using the DFSMSrmmdefault retention period, but the label field is left as zeros.

11—Accessibility (1 byte)

- *Contents:* A code indicating the security status of the data set, as follows:

Valid Version 3 or Version 4 characters: If the volume is not RACF protected, the file access exit is entered.

Space No data set access protection.

1 Password protection. Additional identification of the data set is required before it can be read, written, or deleted. (Ignored if volume is RACF defined.) Password protection can be specified in the PASSWORD subparameter of the LABEL keyword of JCL.

3 Password protection. Additional identification of the data set is required before it can be written or deleted. (Ignored if volume is RACF defined.) This can be specified in the NOPWREAD subparameter of the LABEL keyword of JCL.

Other character Protected volume. No access is possible under the operating system, unless the volume has been defined to RACF and is authorized for use by RACF.

- *Processing:* For input, the system inspects this field on a single volume data set, on each concatenated data set, and on each volume of a multivolume data set. If password protection is specified in this field, the system verifies the password furnished by the operator or TSO terminal user and sets a security indicator in the JFCB. If a valid Version 3 or Version 4 character is specified and the volume is not defined to RACF, the file access exit is entered (the IBM-supplied exit routine will reject the volume). If a character other than an ASCII space, valid Version 3 or Version 4 character, 1, or 3 is encountered, the DCB will not be opened and no further processing will take place.

For output, the system inspects this field in the existing HDR1 label. If 1 or 3 or 4 is specified, with the system code "IBMZLA," the existing data set cannot be overwritten until the system verifies the password and the data set name in Field 3 of this label. Password checking is bypassed if the volume is defined to RACF. If you specify a data set name different from the one in Field 3, and the

|
|

data set is the first one on the first or only volume, the operator must remove the tape and mount a scratch tape, even though you requested a specific volume. If the data set is not the first one on the volume or this is not the first volume of a multivolume data set, the job is abnormally terminated. If a valid Version 3 or Version 4 character is specified and the volume is not defined to RACF, the file access exit will be entered (the IBM-supplied exit routine will reject the volume).

When the system creates labels, the user's request for security is determined from the indicator in the JFCB for password processing or from a JCL ACCODE value. Password codes override ACCODE values if they are both specified.

12—Block Count (6 bytes)

- *Contents:* This field in the trailer label shows the number of data blocks in the data set on the current volume. This field in the header label is always 000000.
- *Processing:* The DCB block count (in the device-dependent area of the DCB) is increased as the data set is read. The final DCB count is compared with the count in the trailer label at end of data or end of volume. If the counts do not agree, a user exit entry in the DCB exit list determines whether processing continues or abnormally terminates. A block count discrepancy causes processing to abnormally terminate if the appropriate user exit entry is not provided.

For read backward, the verification process is reversed. The trailer label count is recorded in the DCB and decreased as the data set is read. The final DCB count should be 0, which is equal to the count in the header label.

When the system creates labels, the block count in the header label is set to 000000. The block count in the trailer label is obtained from the DCB during close and EOVS label creation.

If the data set was created with an EXCP DCB that had no device-dependent area, and the volume was not rejected, the block count is written as zero and is not verified. For tapes with Version 3 labels, a data set opened without at least a 4-word device-dependent area in the DCB will cause the label validation exit to be entered with a symmetry error, unless validation has been suppressed. The default of the exit is to reject the volume.

13—System Code (13 bytes)

- *Contents:* A unique code, IBMZLA, that identifies the system.
Note: Version 1 tapes produced by MVS contain OS/360 or OS/370 as a system code.
- *Processing:* On input, the field is checked to determine how succeeding labels are to be processed (the field determines whether Field 3, "Record Format," and Field 6, "Reserved for Operating System," in the second header label will be processed). On output, the operating system supplies the IBMZLA code.

14—Reserved for Future Standardization (7 bytes)

- *Contents:* Reserved for possible future use; contains blanks.
- *Processing:* Not used or verified, except to check for blanks. When creating labels, the system writes blanks in this field. Blanks are translated to ISCI/ASCII space characters on output.

Format of ISO/ANSI Data Set Label 2 (HDR2/EOV2/EOF2)

Figure 18 on page 82 shows the format of HDR2, EOVS, and EOF2. The shaded areas represent fields that the operating system writes in the label, but are not used or verified during processing. The contents and processing of each field of the label are described, as are differences between ISO/ANSI labels and IBM labels.

If the labels are produced by MVS, they are treated like IBM standard header 2 and trailer 2 labels. If the labels are produced by another system, the “Reserved for Operating System” and “Record Format” fields are not used.

Position	(Bytes)	Field Number and Name
1	(3)	1 Label Identifier
3		
4	(1)	2 Label Number
5	(1)	3 Record Format
6	(5)	4 Block Length
10		
11	(5)	5 Record Length
15		
16	(35)*	6 Reserved for Operating System
50		
51	(2)*	7 Buffer Offset
52		
53		Functional field. Field with no processing function.
	(28)	8 Reserved for Future Standardization
79		
80		

* ISO/ANSI field differs from corresponding IBM field.

Figure 18. Format of ISO/ANSI Header 2 and Trailer 2 labels

1—Label Identifier (3 bytes)

- *Contents:* Three characters that identify the label are as follows:

HDR	Header label (at the beginning of a data set)
EOV	Trailer label (at the end of a tape volume, when the data set continues on another volume)
EOF	Trailer label (at the end of a data set).

- *Processing:* The system checks this field to verify that the record is an ISO/ANSI data set label.

For input data sets, the system checks the label identifier to determine whether data set processing should continue. When the system finds an EOV label, it performs volume switching. When the system finds an EOF label, it passes control to the user's end-of-data routine or continues processing with a concatenated data set.

If the DD statement specifies OPTCD=B for an input data set, the system accepts either EOV or EOF as the trailer label identifier, and the identifier is not used to determine whether a volume switch is necessary. If more volumes are available, the system performs the switching. If no volumes are available, the system passes control to the user's end-of-data routine or continues processing with a concatenated data set.

When creating trailer labels, the EOV routine writes EOV in this field, and the close routine writes EOF.

2—Label Number (1 byte)

- *Contents:* The relative position of this label within a set of labels of the same type; it is always 2 for data set label 2.
- *Processing:* Verified and written in conjunction with Field 1 to identify this label as HDR2, EOV2, or EOF2.

3—Record Format (1 byte)

- *Contents:* An alphabetic character that indicates the format of the records in the associated data set:

F	Fixed length
D	Variable length
S	Spanned.

Note: RECFM=U (undefined length) is accepted for input from a Version 1 tape. If specified for input or output for Version 3 or Version 4 tapes, the label validation exit is entered.

For detailed information about record formats, see *DFSMS/MVS Using Data Sets*.

- *Processing:* For input, the open routine obtains the record format from this label, converts it to a bit code, and records it in the JFCB (if the JFCB field is 0). Then the record format in the JFCB is recorded in the DCB (if the DCB field is 0). Note that this is a merging process in which existing specifications in the JFCB and DCB cannot be overridden.

Record format will not be accepted from the label if the block attribute (in Field 6) is not applicable to MVS; in this case, record format must be specified in JCL or in the DCB.

When creating labels, the system's OPEN and EOVS routines do a reverse merge. The record format in the DCB overrides the record format in the JFCB, and the updated JFCB provides the information for the label.

This merging process is explained and illustrated in Figure 2 on page 4.

4—Block Length (5 bytes)

- *Contents:* A number from 18 to 2048 for Version 3 or from 18 to 32760 for Version 4 indicates the block length (including buffer offset and padding) in bytes.
- **Note:** The 2048-byte limit on block length is part of the Version 3 standard. Larger blocks (up to 9999 bytes for format D and format S records and 32760 bytes for format F records) may be specified with the agreement of the interchange parties. However, for tapes with Version 3 labels, exceeding the 2048-byte limit causes the label validation exit to be entered. The maximum block length for Version 4 is 32760 bytes.

The system can determine the optimum block size when creating tape data sets. For more information, see *DFSMS/MVS Using Data Sets*: "System-Determined Block Size."

Interpretation of the number depends on the associated record format in Field 3, as follows:

Format F	Maximum block length. If the block attribute in field 6 contains a blank, this field is also the minimum block length.
Format D	Maximum block length that includes the 4-byte length indicator in the records and the optional block prefix.
Format S	Maximum block length that including the optional block prefix, plus one or more pairs of 5-byte segment control words and segments.

- *Processing:* The number in the label is converted to binary and merged with appropriate fields in the JFCB and DCB. The merging process is the same as that for the record format code in Field 3 of this label.

5—Record Length (5 bytes)

- *Contents:* A number that indicates the record length in bytes. Interpretation of the number depends on the associated record format in Field 3, as follows:

Format F	Actual record length
Format D	Maximum record length that includes the 4-byte length indicator in the records.
Format S	Maximum record length that excludes all the 5-byte segment control words that describe the record. If the record length is larger than 99999, this field is 0.

- *Processing*: The number in the label is converted to binary and merged with the appropriate fields in the JFCB and DCB. The merging process is the same as for the record format code in Field 3 of this label.

6—Reserved for Operating System (35 bytes)

- *Contents*: If the label is produced by this operating system (the system code in the first header label is IBMZLA), the content and format of this field are similar to the content and format of the seven fields that follow the record length field in an IBM standard label. If the label is produced by another system, the content and format of this field are optional, and the field is not processed by the operating system. The fields produced by the operating system are:
 - Tape Density (1 byte)
 - *Contents*: A code indicating the recording density of the tape; the code is equivalent to the DEN parameter value on the DD statement (for the DEN parameter values, refer to “Tape Characteristics” on page 12). If DCB=DEN was not specified in JCL, this field is set to blanks.
 - *Processing*: If DCB=DEN was not specified in JCL, this data is merged in the JFCB for input. When the system creates labels, the information for this field is obtained from the JFCB.
 - Data Set Position (1 byte)
 - *Contents*: A code indicating a volume switch is as follows:

0	No volume switch has occurred.
1	A volume switch previously occurred.
 - *Processing*: Not used or verified. When creating labels, the open routine writes 0 in this field, and the EOVS routine writes a 1. The close routine determines which code to write by comparing the first volume serial number in the JFCB to the number in the UCB. The close routine writes 0 if the numbers are equal, and 1 if they are not equal.
 - Job/Job Step Identification (17 bytes)
 - *Contents*: Identification of the job and job step that created the data set. The first 8 bytes contain the name of the job; the ninth byte is a slash (/), and the last 8 bytes contain the name of the job step.
 - *Processing*: Not used or verified. When the system creates labels, it sets this field as described above.
 - Tape Recording Technique (2 bytes)
 - *Contents*: Recorded as blanks for all tapes except 7-track tapes. A 9-track tape can only be recorded in odd parity with no translation.
 - *Processing*: Not used or verified.
 - Control Characters (1 byte)
 - *Contents*: A code indicating whether a control character set was used to create the data set, and the type of control characters used:

A	Contains Version 3 or Version 4 control characters.
b	Contains no control characters.

- *Processing*: The specification in the label is converted to a bit code and merged with the Record Format field in the JFCB if ANSI carriage control was not specified as part of DCB=RECFM in JCL.
- Buffer Alignment Block (1 byte)
 - *Contents*: Reserved for future use; recorded as blanks
 - *Processing*: Not used or verified. When creating labels, the system writes blanks in this field.
- Block Attribute (1 byte)
 - *Contents*: A code indicating the block attribute used to create the data set:

B	Blocked records
b	Records not blocked.
 - *Processing*: The specification in the label is converted to a bit code and merged with the record format field in the JFCB. The merging process is the same as for the record format code in Field 3 of this label.
- Coded Character Set ID (CCSID) (5 bytes)
 - *Contents*: A code indicating the CCSID used to create the user data or the Version 4 tapes or blank.
 - *Processing*: The CCSID used to convert data to and from the tape. If blank, the data is written in ISCI/ASCII characters.
- Reserved (6 bytes)
 - *Contents*: Recorded as blanks.
 - *Processing*: Reserved for definition by IBM. Currently not used or verified.

7—Buffer Offset (2 bytes)

- *Contents*: The length of the block prefix (from 0 to 99).
- *Processing*: Used to determine the length of an optional prefix that may be a part of a physical block on tape. The version of the prefix for variable and spanned record formats is known as a block descriptor word (BDW). A BDW is always 4 bytes long and contains the block length of the physical record it describes, including the BDW. The BUFOFF=L operand informs the system that the prefix is an MVS BDW. The prefix is not made available as part of the data read into storage by the queued access method. For more information about BDWs, see *DFSMS/MVS Using Data Sets*.
- *Difference from IBM Field*: This field is not present in IBM standard labels.

8—Reserved for Future Standardization (28 bytes)

- *Contents*: Reserved for possible future use; recorded as blanks. The blanks are translated to ASCII space characters on output.
- *Processing*: Not used or verified. When creating labels, the system writes blanks in this field.

| Format of ISO/ANSI User Header and Trailer Labels (UHL/UTL)

Figure 19 shows the format of UHL and UTL labels.

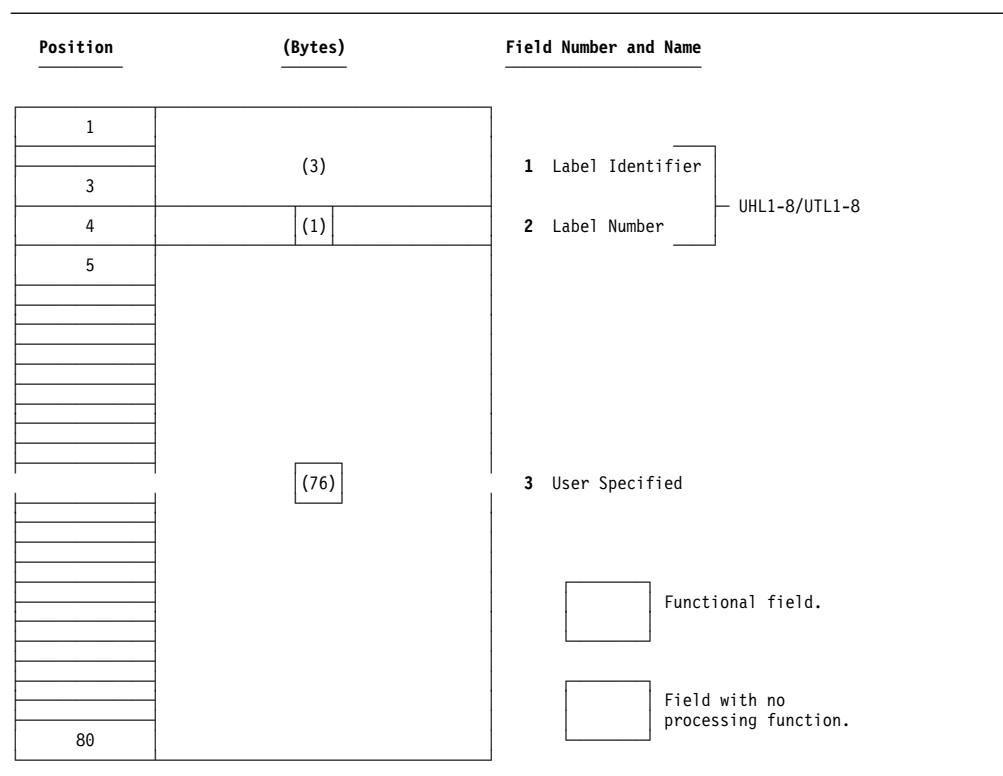


Figure 19. Format of ISO/ANSI User Labels

1—Label Identifier (3 bytes)

- *Contents:* Three characters that identify the label are as follows:

UHL User header label (at the beginning of a data set)

UTL User trailer label (at the end-of-volume or end-of-data set).

- *Processing:* This field is read to verify that the record is a user label. The system accepts either UHL or UTL.

2—Label Number (1 byte)

- *Contents:* Any valid Version 3 or Version 4 character.
- *Processing:* This field is checked by the operating system for a valid Version 3 or Version 4 character during creation of a user header label (UHL) during open/EOV if validation has not been suppressed. If an invalid character is detected, the label validation exit is entered.
- *Difference from IBM Field:* This field can contain only numeric 1 to 8 for IBM standard user labels. A maximum of 8 user header or trailer labels is supported for conventional IBM standard user labels, but any number of user labels can be written for Version 3 or Version 4 tapes, and they may be lettered or numbered in any order.

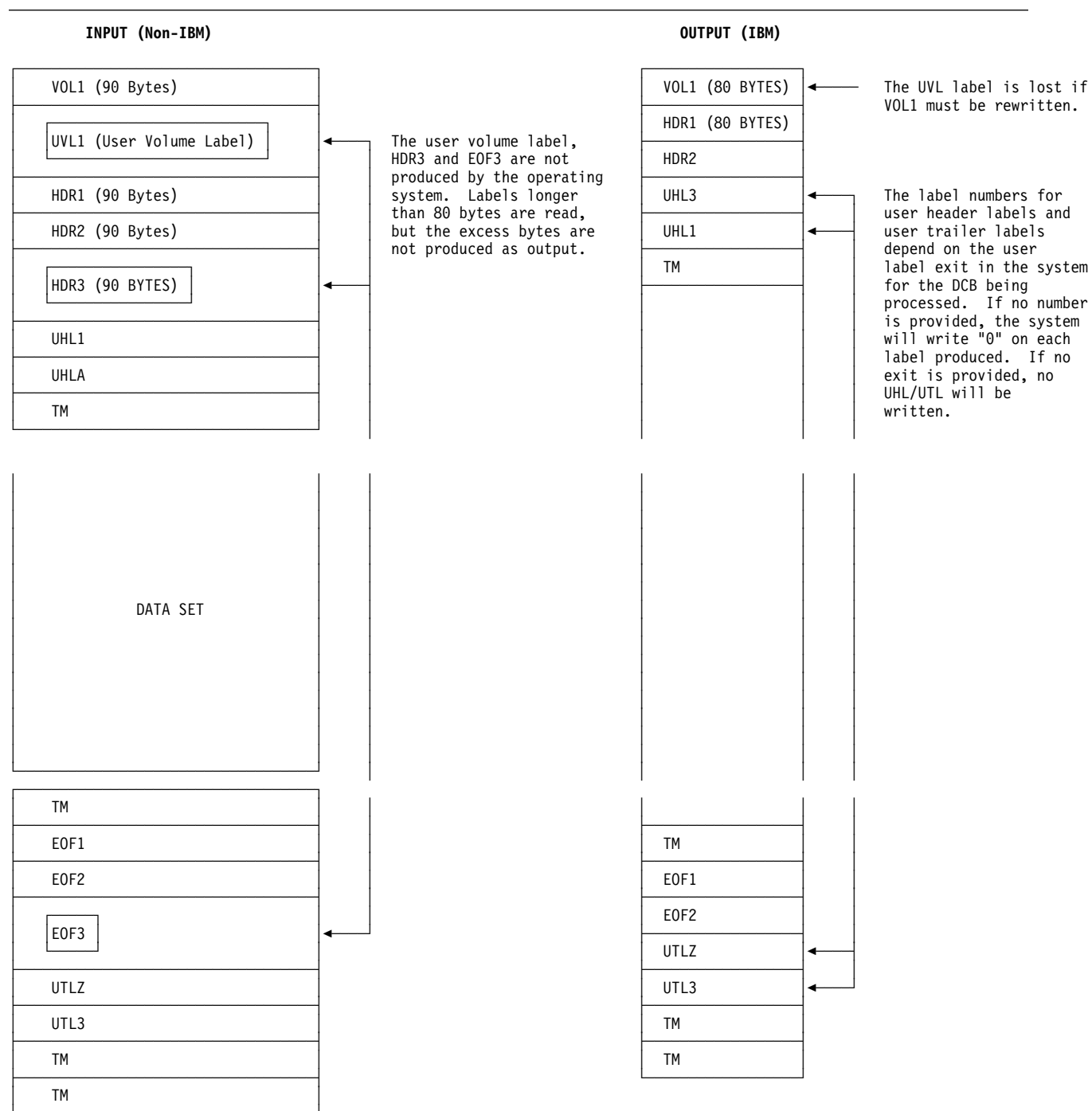
3—User Specified (76 bytes)

- *Contents*: Specified by the user, but must be valid Version 3 or Version 4 characters.
- *Processing*: Specified in the DCB exit list. This field is checked as explained for Field 2.

Other File Header Labels

Other file header labels (HDR3-HDR9, EOF3-EOF9, EOV3-EOV9) and user volume labels (UVL1-UVL9) and volume labels (VOL2-VOL9) will not be created by the operating system. Because these labels may appear on magnetic tapes created by other systems, the operating system will accept them as input. Those labels are ignored during label processing and are not placed on output tapes. Ignored labels are checked only for a valid label identifier and proper label sequence; no checks are made for ignored labels encountered during positioning to the requested data set.

Figure 20 on page 89 shows a hypothetical input tape from a non-IBM system and a corresponding output tape produced by the operating system. The user volume label, the additional header label, and the additional trailer label are not placed on the output tape.



Corresponding Output Tape

Figure 20. Example of a ISO/ANSI Tape from a Non-IBM System and a Corresponding Output Tape

Overview of ISO/ANSI Label Processing

Label processing is handled by the I/O support routines of data management (open, EOV, and close). This processing consists of the following basic functions:

- Translating input labels from ASCII to EBCDIC and translating output labels from EBCDIC to ASCII
- Checking the labels on input tapes to
 - Ensure that the correct volume is mounted
 - Identify, describe, and protect the data set being processed
 - Attempt to ensure maximum correctness and consistency of data sets and their labels
 - Identify compatibility conflicts with Version 3 or Version 4 standards
- Checking the existing labels on output tapes to
 - Ensure that the correct volume is mounted
 - Prevent overwriting of vital data
 - Identify compatibility conflicts with Version 3 or Version 4 standards
- Creating and writing new labels on output tape
- For volumes in an automated tape library, comparing the external volume serial number to the magnetically recorded internal label.

These processing functions are summarized in Figure 21. The table shows the specific labels that are processed for each function and which routines perform the functions.

Although the default of each IBM-supplied installation exit is to reject the volume, the exits can be modified to do additional label processing. See “ISO/ANSI Installation Exits” on page 99 for more information.

All volumes will be written with Version 3 or Version 4 labels. For information about using tapes with other than Version 3 or Version 4 labels, see “Compatibility with Non-Version 3 or Non-Version 4 Volumes” on page 102.

Figure 21. ISO/ANSI Label Processing by Open/Close/EOV Routines

	Volume Label			Header Labels ¹				Trailer Labels ¹			
	VOL1	VOL 2–9	User Volume Labels	HDR1	HDR2	HDR3-9	UHL	EOF1 or EOV1	EOF2 or EOV2	EOF3-9 or EOV3-9	UTL
Processing											
First or only volume ²											
Checks labels on input tape	Open	Ignored ³	Ignored ³	Open	Open	Ignored ³	Open	EOV	Bypassed	Ignored ³	EOV
Checks existing labels on output tape before over-writing	Open	Ignored ⁴	Ignored ⁴	Open	Not read	Not read	Not read	Not read	Not read	Open	
Writes new labels on output tape	Open or user ⁵	Not written	Not written	Open	Open	Not written	Open	Close or EOV	Close or EOV	Not written	Close or EOV
Compares the volume serial number on the external volume label to the serial number in the tape volume label ⁷	Open	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸
Second or subsequent volumes ⁶											
Checks labels on input tape.	EOV	Ignored ³	Ignored ³	EOV	Bypassed	Ignored ³	EOV	EOV	Bypassed	Ignored ³	EOV
Checks existing labels on output tape before over-writing.	EOV	Ignored ⁴	Ignored ⁴	EOV	Not read	Not read	Not read	Not read	Not read	Not read	Not read
Writes new labels on output tape.	EOV or user ⁵	Not written	Not written	EOV	EOV	Not written	EOV	Close or EOV	Close or EOV	Not written	Not written
Compares the volume serial number on the external volume label to the serial number in the tape volume label ⁷	EOV	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸	N/A ⁸
Notes:											
1. For read backward operations, the action on header and trailer labels is reversed. 2. Includes the first volume of concatenated data sets with unlike characteristics when the user program has turned on the DCB unlike attributes bit (DCBOFPPC). See “Concatenating Unlike Data Sets” in <i>DFSMS/MVS Using Data Sets</i> . 3. Label sequence is checked but contents are ignored. 4. Operator must give Open/EOV permission to overwrite VOL2–9 or ULV labels. 5. User can create the label with the EDGINERS or IEHINITT utility programs, or a user program. 6. Includes the first volume of concatenated data sets when the user has not turned on the DCB unlike attributes bit. 7. Applies only to Tape Library Dataserver. 8. Not Available.											

Creating a Volume Label

The VOL1 label is usually created by the EDGINERS or IEHINITT utility programs, or a user's program when the volume is first received at the installation. At that time, a permanent volume serial number is assigned to the volume, physically posted on the volume, and recorded in the VOL1 label.

The IBM-supplied utility programs EDGINERS or IEHINITT write the following labels in ASCII code:

1. A VOL1 label with the volume serial number, accessibility code, and owner identification that you specify. You cannot specify any other fields of the VOL1 label.
2. A dummy HDR1 label with '0001' in the file section number, file sequence number, and generation number fields; "IBMZLA" in the system code field; and a space in the accessibility field (allowing unlimited access). Reserved fields contain zeros, with a leading space in the date fields.
3. A tapemark.

A tape initialized with these labels should not be confused with an Version 3 or Version 4 tape, which requires one data set (the data set can be empty).

Version 3 or Version 4 standards require label symmetry around an empty data set when a volume is initialized. The labels written by EDGINERS and IEHINITT are accepted by the system routines that produce symmetrical labels. The HDR1 label is updated with system data, the single tapemark is overwritten with a HDR2 label containing data set characteristics, and a tapemark is written to complete the beginning-of-volume and beginning-of-file label groups. A set of trailer labels are written when the data set is closed, even if no data was written in the data set.

The IEHINITT utility program can write a VOL1 label on a labeled, unlabeled, or blank tape; it makes no checks to see what data, if any, exists on the tape. Therefore, IEHINITT does not check for password or RACF security protection; it does not create, modify, or delete RACF profiles of RACF-defined volumes. IEHINITT can create either Version 3 or Version 4 tape labels. Detailed procedures for using the program are described in *DFSMS/MVS Utilities*.

The EDGINERS utility program validates the mounted volume before writing a VOL1 label on a labeled, unlabeled, or blank tape. DFSMSrmm does not check for password or RACF protection, but can be used to create RACF profiles for new volumes. It can also be used to create Version 3 or Version 4 tape labels. For details on how to do this, see *DFSMS/MVS DFSMSrmm Implementation and Customization Guide*.

Methods other than the EDGINERS or IEHINITT utility programs can be used to write VOL1 labels. You can use a card-to-tape program, or you can replace the IBM-supplied volume label editor routine with one that writes VOL1 labels. See "ISO/ANSI Volume Label Verification and Editor Routines" on page 94 for more information. Some data or a tapemark should already exist on the tape; otherwise, the tape control unit may read through the entire reel of blank tape looking for some data.

For all IBM magnetic tape subsystems, the VOL1 label is rewritten by the open or EOVR routines if **all** of the following conditions are met:

- A density conflict has occurred or ISO/ANSI routine conflict has occurred.
- OUTPUT, OUTIN, or OUTINX is specified in the OPEN macro.
- The tape is positioned to the first data set on the volume.
- Either of these conditions are true:
 - The data set is not password protected
 - The volume is RACF protected, the system-wide RACF tape protection option has been specified, and the user is ALTER authorized.
- The volume does not have VOL2–9 or UVL labels. If it does, the operator must give permission to rewrite the VOL1 label because the VOL2–9 and UVL labels will be overwritten.

If you request output to the first data set on a Version 3 or Version 4 output volume (AL, AUL), and the tape that you are allocated is recorded in the wrong density and cannot be read or in the wrong ISO/ANSI version level, then the open or EOVS routine can rewrite the VOL1 label in the density or version you specify.

This allows you to make nonspecific requests for output tapes and allows the operator to mount any available scratch tape. However, if the system-wide RACF tape protection option has been specified, then the volume is rejected.

If you request output to any data set other than the first data set on the volume and a density or a version conflict occurs, then all further processing is done in the density or version of the existing VOL1 label (unless the IBM-supplied volume label editor routines are replaced with installation-written routines that do otherwise).

Note: If the densities are mixed, the tape will not meet the specifications of Version 3 or Version 4 standards.

If you request an output volume for an Version 3 or Version 4 labeled (AL, AUL) tape and the mounted volume is an NL, NSL, SL, or SUL tape, then the open or EOVS routine creates a VOL1 label after checking authorization for access and the expiration date. The routine sends a message to the console operator requesting serial number and owner information (see Figure 16 on page 72, Fields 3 and 7).

The system programmer can indicate whether to create a Version 3 or Version 4 label by specifying a parameter in the DEVSUPxx PARMLIB member. For details see *OS/390 MVS Initialization and Tuning Reference*.

The system uses the following hierarchy to determine which version of labels will be written when opening the first file of the first volume of an ISO/ANSI tape data set for output:

- Maintain version level in VOL1 label if output DISP=MOD, or for EXTEND, or if on the first data set as the volume.
- Maintain version level on VOL1 label if extending data set to a new volume.
- Use version level specified in the volume mount installation exit. For details, see *DFSMS/MVS Installation Exits*.
- Use the specified 'Forced' version level in the DEVSUPxx member. For details, see *OS/390 MVS Initialization and Tuning Reference*.
- Use the existing version level as specified in the VOL1 label of Version 3 or Version 4.

- Use the specified version level in the DEVSUPxx member. For details, see *OS/390 MVS Initialization and Tuning Reference*.
- Use Version 3 as the default.

ISO/ANSI Volume Label Verification and Editor Routines

If you specify that an input or output tape has a Version 3 or Version 4 label, the operating system checks for the Version 3 or Version 4 volume label at the beginning of the tape.

Because of conflicting label types or conflicting tape characteristics, various error conditions can occur during the verification of the first record. Under some error conditions, the tape is accepted for use. Under other error conditions, the tape is not accepted, and the system issues another mount message. For certain other error conditions, the system gives control to a label anomaly exit routine and to a *volume label editor routine*; your installation can use IBM-supplied routines or it can supply its own routines.

The IBM-supplied volume label editor routines determine the discrepancies between the requested tape and the mounted tape and, if necessary, pass control to the appropriate system routine to create or destroy labels. If DFSMSrmm is controlling the volume, the DFSMSrmm label editor routine controls the creation or destruction of standard labels using RACF profiles in the FACILITIES class. Installation-supplied routines can perform other functions.

IBM recommends using tape management exit routines instead of the volume label editor routines.

For detailed information about these and other available exit routines, see *DFSMS/MVS Installation Exits*.

Protecting Data

The accessibility fields in the VOL1 and HDR1 labels indicate whether a volume and data set are protected against unauthorized use. Version 3 or Version 4 volumes can be protected by means of one of the following:

- RACF
- IBM-supplied Version 3 or Version 4 installation exits described in "ISO/ANSI Installation Exits" on page 99. These exits can be replaced by installation-written exit routines.
- Data set password protection.

Note: In a IBM system-managed library, data set password protection is not supported.

- A combination of the installation exits and password protection.

Version 1 input volumes are protected by either RACF or data set password protection.

Note: All checking for authorization is bypassed if security processing is suppressed. This can occur, for example, when the program properties table is marked to suppress security checking.

Only the system programmer can update the program properties table. For information about the program properties table, see *OS/390 MVS Initialization and Tuning Reference*.

RACF allows you to establish access requirements for both tape data sets and tape volumes. To protect data on tape, you can do either or both of the following:

- Control access to the tape volume(s).
- Control access to individual tape data set(s) on the tape volume(s).

RACF protection at the volume level overrides RACF protection at the data set level. For more information on how to activate these levels of RACF protection and how they interact with each other and with your own tape management system, see the *OS/390 Security Server (RACF) Security Administrator's Guide*. DFSMSrmm supports RACF protection, but not password protection. For information about DFSMSrmm and RACF, see *DFSMS/MVS DFSMSrmm Implementation and Customization Guide*.

The following principles apply to RACF protection at the volume level:

- ALTER access authority is required to create or destroy the VOL1 label.
- READ access authority is required to open the volume for input (open options INPUT or RDBACK).
- UPDATE access authority is required to open the volume for output (open options OUTPUT, EXTEND, INOUT, OUTIN, or OUTINX).

The user can open the volume to read or write if the tape volume is defined to RACF, the user has UPDATE access authority, and PROTECT=YES has not been specified in the JCL.

The request fails if the tape volume is defined to RACF, the user has UPDATE authority, PROTECT=YES has been specified in the JCL, and the tape is not a RACF scratch volume.

If the tape volume is defined to RACF and the user has READ but not UPDATE access authority, or if the user has UPDATE access but PROTECT=YES has been specified in the JCL and the volume is a RACF scratch tape volume, the system does not grant the user access to read until it has ensured that the user can not write on the tape. The user cannot access the volume until one of the following conditions is met:

- **Hardware Protection.** If the write-enable ring has been removed from the tape reel or the write-protect tab has been set to disable writing on the tape cartridge, the tape volume cannot be written on. The system safely permits the user to access the tape to read. This hardware protection cannot be circumvented by software.
- **Logical write-protection.** If the write-protect tab on an IBM magnetic tape cartridge is set to enable writing, the system issues a hardware command to prevent writing on that cartridge. If the command succeeds, the system safely allows the user to access the tape to read. An unauthorized program cannot bypass this combination of hardware and software protection.
- **IEC.TAPERING.** Your installation may choose to depend on a tape management system to prevent accidentally overwriting tapes with unexpired data. Typically, a tape management system only allows volumes with no unexpired data

to be opened for output. DFSMSrmm provides facilities to prevent accidental overwriting of non-scratch tape volumes. The IEC.TAPERING support facilitates the operation of tape management systems because it allows all volumes to remain write-enabled (by the ring in the volume or the switch on the cartridge). This eliminates the need for operator intervention.

If the write ring or the cartridge tab is set to enable writing, the system checks if the user is authorized for read to the IEC.TAPERING profile in the RACF FACILITY class. If the user does have this authority, the system grants the user access.

Attention: If you use the IEC.TAPERING support to allow users to read from tapes that are enabled for writing, when the users are only authorized to read, the system software cannot prevent knowledgeable users from writing on any files on the tapes.

- **Operator Intervention.** If none of the preceding conditions are met, the system requires the operator to intervene and prevent writing on the volume. The system demounts the tape and issues a message asking the operator to remove the write-enable ring from the tape reel or change the switch on the tape cartridge. After the operator remounts the tape, the system continues to protect the volume from unauthorized writing by repeating the preceding checks, beginning with the check for hardware protection. This check continues until one of the previously mentioned conditions is met.

If the tape volume is not defined to RACF, access is granted and processing continues. For an overview of RACF protection for tape volumes, see *OS/390 Security Server (RACF) Security Administrator's Guide*. For information on how DFSMSrmm can help you manage RACF security for your tape volumes, see *DFSMS/MVS DFSMSrmm Implementation and Customization Guide*.

For more information on data set password protection, see *DFSMS/MVS DFSMSdfp Advanced Services*.

Volume Accessibility

Version 3 or Version 4 Volumes: If the volume is RACF protected, and the volume security processing has not been suppressed, the RACHECK installation exits are entered to accept or reject the volume (by a return code from RACHECK). If the VOL1 accessibility field contains a valid Version 3 or Version 4 character, the RACHECK parameter list is initialized to address the ISO/ANSI exit parameter list (IECIEPRM) and the accessibility code. Otherwise, the RACHECK installation exits are passed zeroed pointers for the ISO/ANSI parameters. If RACF accepts the volume but the ISO/ANSI parameters to the exit are zero, the validation suppression exit is entered.

If RACF is not available, or the volume is not defined to RACF, OPEN/EOV checks the accessibility field in the VOL1 label. If the field contains a valid Version 3 or Version 4 character, the volume access exit is entered to accept or reject the volume (by a return code in the exit parameter list). If the field contains a space, the validation suppression exit is entered. All other characters cause the volume to be rejected.

The VOL1 accessibility code can be changed when the VOL1 label is changed by the EDGINERS or IEHINITT utility programs or by the operator during open. See "Volume Label" on page 68.

Version 1 Volumes: First, OPEN/EOV checks the accessibility field in the VOL1 label. If the field contains a space character, the volume is defined to RACF, and volume security processing has not been suppressed, then the RACHECK installation exits are entered to accept or reject the volume (by a return code from RACHECK). Access to the volume is allowed if the field contains a space character, but the volume is not defined to RACF, or the volume security processing has been suppressed. Each data set on the volume can be individually protected. The volume is rejected if the field contains anything other than a space character.

The VOL1 accessibility code can be changed when the VOL1 label is changed by the EDGINERS or IEHINITT utility programs, or by the operator during open. See "Volume Label" on page 68.

Data Set Accessibility

If a volume is RACF protected, and access to the volume was checked when the volume was verified (by RACHECK processing), then data set accessibility fields are not checked; therefore, any data set on the volume can be accessed. If a volume is not RACF protected, then the system inspects the accessibility field in the HDR1 label of the requested data set. For a multivolume data set, the system inspects the HDR1 label on each volume. The accessibility codes of any concatenated data sets are also inspected.

Version 3 or Version 4 Volumes: 1 or 3 in the data set accessibility field with a system code of "IBMZLA" indicates an MVS password-protected data set; a space character in the accessibility field allows unlimited access; and a valid Version 3 or Version 4 character causes the file access exit to be entered for authorization processing. The file access exit is also entered when a new data set is added to an output volume and if the first character of the JCL ACCODE keyword is a valid character. Any other character causes the volume to be rejected.

If the accessibility field of a data set being opened contains a valid character or a space, no attempt is made to check the accessibility field of any other data sets that may exist on the volume; therefore, the other data sets can be overlaid by new output. If accessibility checking for multiple data sets is required, it must be done by an installation-written file access exit. Never place an access-protected data set on the same volume with unprotected data sets because no exit is entered for unprotected data sets.

An HDR1 accessibility code of valid character can be specified by the ACCODE parameter of JCL. ACCODE can also be specified for dynamic allocation (SVC 99 key). Changes to an HDR1 accessibility code of a valid character can be monitored by an installation-written file access exit. Password codes override ACCODE values if both are simultaneously specified.

Version 1 Volumes: 1 or 3 in the data set accessibility field indicates a password-protected data set.

A space character in the accessibility field allows unlimited access, and no attempt is made to check the accessibility field of any other data sets that may exist on the volume; therefore, the other data sets may be overlaid by new output.

Input Processing for Password-Protected Data Sets

The system asks the operator or TSO user to enter the correct password for input processing of a password-protected data set. The password is verified in a user-established password data set. This password data set contains the data set name, the password, and a protection-mode indicator. The data set name for Version 3 generation data sets does not include the .GxxxxVyy suffix. The protection-mode indicator is set to permit either read/write or read-only operations. Processing is terminated if:

- The operator or TSO terminal user does not supply the correct password in two attempts.
- The password record for the data set to be opened does not exist in the password data set.

DFSMS/MVS DFSMSdfp Advanced Services describes the protection feature and discusses how to create and maintain the password data set.

Note: Password protection for a tape data set in an IBM automated tape library, or manual tape library, is not supported. In such tape libraries, the only security protection for the tape data set is via RACF.

Output Processing for Password-Protected Data Sets

The system compares the data set name shown in the HDR1 label with the data set name specified in the DD statement for output processing of a password-protected data set. For generation data sets on a Version 3 volume, the comparison of names does not include the generation and version numbers.

If the names are not the same, processing is terminated unless the data set is the first one on the first or only volume. In this case, even if you specify a specific volume, the operator must demount the tape and mount a new scratch tape. If the names are the same, the system requests the operator or TSO user to key in the required password. The password is verified in a user-established password data set in the same manner as described above for an input data set. The read/write protection mode is necessary for output data sets.

Two restraints are placed on creation of password-protected data sets:

- When creating a password-protected data set following an existing password-protected data set, supply the password of the existing data set. The accessibility code of "1" or "3" must be the same in both the existing and the new data set. This is true even if the volume is RACF defined, or resides in an IBM automated tape library or a manual tape library.
- When creating a multivolume, password-protected data set, the second and successive volumes will also be verified. Verification consists of ensuring that the data set name in the JFCB (for Version 3 volumes, not including generation and version numbers, if present) is the same as the data set name in the password record. Verification also ensures that the protection-mode indicator allows writing to the data set. This verification is bypassed if the volume resides in an IBM automated tape library or a manual tape library.

Deleting a Password-Protected Data Set: If a password-protected data set is deleted, the HDR1 accessibility field must be set to a space character before the volume can be written on again. The HDR1 accessibility field can be set by using either the EDGINERS or IEHINITT utility programs, or a user program to relabel the volume.

ISO/ANSI Installation and Use Characteristics

This section summarizes the main characteristics of installing support for and using ISO/ANSI Version 3 or Version 4 volumes.

ASCII Conversion

The IBM-supplied conversion routine converts ASCII 7-bit code and is designed to support Version 3 or Version 4 standards. The system forces OPTCD=Q during open to cause ASCII conversion during processing of any volume mounted as LABEL=AL.

Note: If the input data includes raw binary, floating point, or packed decimal fields, then any bytes containing values that convert into more than 7 bits are converted to X'1A'. This results in a permanent loss of data.

Additional information on ASCII 7-bit code and tables showing the relationship of EBCDIC and ASCII 8-bit code is given in Appendix E, "Equivalent ASCII and EBCDIC Codes" on page 143.

CCSID Conversion

Version 4 allows user data written in any national or user defined character standards. The character standards are specified by the CCSID parameters on the JOB/EXEC and DD statements. For details, see *DFSMS/MVS Using Data Sets*.

Volume Initialization

Version 3 or Version 4 volumes are initialized with the EDGINERS or IEHINITT utility programs, or by the operating system in the case of certain label conflicts during mount verification of a volume. Volume initialization is discussed under "Creating a Volume Label" on page 92.

ISO/ANSI Installation Exits

Four *installation exits* are provided, as defaults, for a Version 3 or a Version 4 volume:

- Volume access
- File access
- Label validation
- Label validation suppression.

A fifth installation exit, WTO/WTOR, can be written (or modified, if one has already been written) by your installation to convert ISO/ANSI non-Version 3 to Version 3 labels.

All the default installation exit routines are supplied in a module containing a single CSECT (IFG0193G, alias IFG0553G), in SYS1.LPALIB. A copy of the source code for the module is contained in member ANSIEXIT of SYS1.SAMPLIB.

The default routines, except the validation suppression exit, reject the volume. You can replace any of the IBM-supplied exit routines with your own, installation-written, exit routines. The default routines can be modified or replaced to perform I/O (such as overwriting a label), change system control blocks, and mount or demount volumes. The return code from the exits can be modified to request continued processing.

For detailed information about these and other available exit routines, see *DFSMS/MVS Installation Exits*.

Volume Access

Version 3 or Version 4 volumes can be protected by either RACF or the *volume access exit*. The volume access exit is entered during open or EOVS if a volume is not RACF protected and the accessibility field in the volume label contains an ASCII valid Version 3 or Version 4 character. The exit is bypassed if volume security checking has been suppressed. See “Protecting Data” on page 94 for more information.

File Access

Version 3 or Version 4 data sets can be protected by the *file access exit*, data set password protection, or a RACF data set profile. The file access exit is entered after positioning to a requested data set if the accessibility field in the HDR1 label contains an ASCII valid Version 3 or Version 4 character and the volume is not RACF protected. Likewise, the exit is entered when a data set is written to an output volume if the first character of the JCL ACCODE is also a valid character. See “Protecting Data” on page 94 for information on data set password protection and RACF.

Label Validation

Labels and program specifications for labels and file structure are checked to ensure that a tape format does not violate Version 3 or Version 4 standards. The label validation exit is entered if any of the following requirements are not met:

- Labels must contain only certain ASCII characters. For a list of the characters, see “Label Definition and Organization” on page 64.
- Label fields must contain properly aligned data. For more information, see “Label Definition and Organization” on page 64.
- Labels that frame a data set must be symmetrical:
 - DISP=MOD is not allowed for extending an existing Version 3 data set or an existing non-IBM format Version 4 data set; however, it can be specified to create a new data set.
 - An open option for EXTEND, OUTINX, or INOUT is not allowed in Version 3 or non-IBM format Version 4.
 - An EXCP DCB used for output must contain at least a 4-word device dependent area.

For more information, see “Data Set Trailer Labels” on page 69, “Creating a Volume Label” on page 92, and *DFSMS/MVS Installation Exits*.

- Block size must not exceed 2048 bytes for Version 3 or 32760 bytes for Version 4. See “Format of ISO/ANSI Data Set Label 2 (HDR2/EOV2/EOF2)” on page 81.
- Variable length records must be specified as Format-D, not as Format-V. Undefined-length records (Format-U) are allowed only for Version 1 input volumes; they are not allowed for Version 3 or Version 4 volumes.
- Duplicate data set names, including names for generation data sets, are not allowed on the same Version 3 volume. See “Processing the HDR1 Label” on page 106.

- Expiration dates for successive data sets on a Version 3 volume must not be in ascending sequence. See “Expiration Date on Existing Label” on page 114.

Label Validation Suppression

The validation suppression exit allows the option of suppressing label validation. It is entered during open or EOVS if:

- Volume security checking has been suppressed
- The volume label accessibility field contains an ASCII space character
- RACF accepts a volume and the accessibility field does not contain a valid Version 3 or Version 4 character.

WTO/WTOR Installation Exit

For ISO/ANSI tape volumes, DFSMS/MVS * 1.1 and above supports output only to Version 3 and input from either Version 1 or Version 3. DFSMS/MVS 1.5 supports output to both Version 3 and Version 4 and input from Version 1, Version 3, or Version 4. If a label version conflict is detected during an output request to the first data set on a volume, the WTOR prompts the installation operator to obtain information for rewriting the volume label as a Version 3 or a Version 4 label. If the installation does not want the operator to provide the necessary label information, the WTO/WTOR installation exit can provide the information.

RACHECK Installation Exits

The RACHECK preprocessing and postprocessing installation exits can act on Version 3 or Version 4 volume accessibility and label validation. These exits are discussed under “Protecting Data” on page 94.

Generation Data Groups

For Version 3 or Version 4, generation and version numbers are *not* part of the data set name, as contained in the file identifier field of the HDR1 label. See “Format of ISO/ANSI Data Set Label 1 (HDR1/EOV1/EOF1)” on page 74.

Spanned Records

Spanned records (Format-S) are allowed and can be up to 16 776 192 bytes long; however, the actual length of logical records to be processed depends upon the amount of virtual storage that can be acquired. See “Format of ISO/ANSI Data Set Label 2 (HDR2/EOV2/EOF2)” on page 81 and *DFSMS/MVS Using Data Sets*.

Block Padding

A fixed-length logical record (Format-F) containing all ASCII circumflex characters (X'5E') will signal an end-of-block condition. A variable-length record (Format-D) beginning with a circumflex character, even though the rest of the record does not contain circumflex characters, signals an end-of-block condition.

Checkpoints

The CHKPT macro service prevents a checkpoint from being taken on as ISO/ANSI tape.

Compatibility with Non-Version 3 or Non-Version 4 Volumes

For input, volumes with Version 1 labels are accepted, but without Version 3 or Version 4 checking. No Version 3 or Version 4 installation exits will get control, and the volume label accessibility code field will not be passed to either the RACF pre-processing or post-processing exits, ICHRCX01 or ICHRCX02. All acceptable version types must have both the labels and data written in ISCII/ASCII 7-bit, unless the IBM-supplied routine IGC0010C is replaced with an installation-written routine.

For output, all volumes will be written with Version 3 or Version 4 labels. A volume with Version 1 labels, or any other volume with an 80-character volume label containing the ASCII characters VOL1 in the label identifier field, might be mounted for output, but only if the first data set is being written. (However, extending an existing data set, such as by specifying DISP=MOD in the DD statement, is not allowed for Version 3 or for non-IBM format Version 4.) The volume label is rewritten and the new header and trailer labels are written in Version 3 or Version 4 format. This converts the volume to Version 3 or Version 4.

Note: This conversion requires intervention by the console operator (in response to message IEC704A during open/EOV). Therefore, with careful operational procedures, you can avoid creating an unexpected Version 3 or Version 4 volume.

Processing Differences Between Version 1 and Version 3 or Version 4

Version 3 or Version 4 processing for certain characteristics differs from Version 1 support. Because of these differences, job streams that run under Version 1 might fail or produce different results when running under Version 3 or Version 4. Therefore, Version 1 users should check their job streams and modify them, if necessary, before installing Version 3 or Version 4 support.

The characteristics that differ Version 3 or Version 4 processing from Version 1 are:

- Label validation
- Generation data groups
- Spanned records
- Block padding
- Checkpoints
- Compatibility with non-Version 3 or non-Version 4 volumes.

Processing Version 3 or Version 4 Volumes on Earlier MVS Systems

When a Version 3 or a Version 4 volume is mounted on an earlier MVS system, the following restrictions apply:

- Input generation data sets are not recognized because the .GxxxxVyy suffix is not part of a Version 3 or a Version 4 file identifier.
- Spanned record format is not supported.
- Data set characteristics recorded in the second header or trailer label of an input volume are not made known to the application program; instead, they must be specified either in JCL or in the DCB. (This is because the Version 3 or Version 4 MVS system code, IBMZLA, is not recognized by earlier systems.)
- A data set having a header label created by a non-MVS system with 1 or 3 in the accessibility field is treated as an MVS password-protected data set unless the volume is defined to RACF.

- A valid Version 3 or Version 4 character in the accessibility field of a header or volume label causes the volume to be rejected.
- Any output does not meet the specifications of Version 3 or Version 4 standards.

Tapemarks

The tapemark requirements for ISO/ANSI interchange tapes are:

- Each header label group must be followed by a tapemark that indicates the beginning of the data to be processed.
- Each data set must be followed by a tapemark that indicates the beginning of the trailer label group.
- If the data set is the last one on the volume, each trailer label group must be followed by two tapemarks. Otherwise, one tapemark is required.

When creating a data set with Version 3 or Version 4 labels, the system routines write the necessary tapemarks.

Opening an Input Data Set

When a data set is opened for input, the volume label, user volume labels, and data set header labels (data set trailer labels for read backward) are processed. User header labels (user trailer labels for read backward) are also processed if they are present, and the user specifies AUL in JCL.

The tape device is checked for a density compatible with the density specified by the user.

Before verifying the volume, the program will be abnormally terminated if the system-wide RACF tape protection option is specified and the DD statement specifies PROTECT=YES without previously opening the DD statement for output processing.

When the unit is ready to read the VOL1 label, the first record on the tape is checked for a length of at least 80 bytes and must contain the ASCII identifier VOL1 in the first 4 bytes. The volume label can be Version 1, Version 3, or Version 4. If the record is over 80 bytes long, the excess characters are ignored.

The system also checks for label conflicts (the type of label requested differs from the type of label read at load point from the mounted volume), compatibility conflicts (the label is not Version 1 or Version 3 or Version 4), and density conflicts (the density specified by the user for the data set is not compatible with the density of the VOL1 label). In the case of a label conflict or version compatibility conflict, the volume is rejected. In the case of a density conflict, the density of the VOL1 label is used for all further processing of the data set.

If system security checking should be bypassed, the validation suppression exit is entered. Otherwise, if the system-wide RACF tape protection option has been specified, RACF access authorization to the volume is checked for READ authority. If a tape with Version 3 or Version 4 labels is not RACF protected and the VOL1 accessibility field contains a valid Version 3 or Version 4 character, the volume access

exit is entered. For more details of RACF and accessibility checking, see “Protecting Data” on page 94.

For a tape with Version 3 or Version 4 labels, (after access to the volume is authorized, the request is checked for possible symmetry violations). Because INOUT can result in asymmetrical labels, specifying this option will cause the label validation exit to be entered (unless it is Version 4 or validation has been suppressed). In addition, unless validation has been suppressed, the VOL1 label is checked for conditions not allowed by Version 3 or Version 4 standards. If an invalid condition is found, the label validation exit is entered.

Volume Serial Number

The system uses the VOL1 label to ensure that the correct tape is mounted. The volume serial number in the label is compared to the volume serial number you specify. Specify the serial number either directly in the DD statement or indirectly through the catalog facility. Serial numbers are required when the processing method is INPUT or RDBACK, EXTEND or OUTINX.

Serial numbers also are required when the processing option is OUTPUT or OUTIN with DISP=MOD.

If the volume serial number is correct, the volume is considered to be mounted and verified. If the serial number is not correct, the system rejects the tape and issues another mount message.

Positioning the Volume to the Data Set

When the volume is mounted and verified, the system positions the tape to the front of the header label group of the data set to be processed. Usually, there is only one data set on the volume, and the header label group immediately follows the volume label.

Unless the data set is cataloged, you specify a data set sequence number in the LABEL parameter of the DD statement to retrieve a data set when more than one data set is on a single volume of tape. You need not specify a data set sequence number for a cataloged data set, because the system obtains the number from the catalog along with the volume serial number.

The sequence number can be from 1 to 9999, with 1 representing the first data set on the volume. If you specify a sequence number higher than the number of data sets on the volume, your task will be abnormally terminated or, if the volume ends with EOVS labels, the open routine will switch to the next volume.

If the data set is not cataloged and you do not specify a sequence number, or you specify 0, the system assumes that the data set is the first in sequence on the volume.

To position the tape, the system uses the requested data set sequence number shown in the JFCB and the data set sequence number shown in the first HDR1 label on the tape, and maintains a logical data set sequence number in the UCB. The number in the UCB represents the current position of the tape, and is maintained as follows:

1. When a tape is first mounted, the data set sequence number in the UCB is 0.

2. When a data set is opened, the open routine sets the data set sequence number in the UCB to 1. The exceptions are:
 - If the tape is still positioned from previous processing, such as for a LEAVE request, the open routine does not reset the number in the UCB.
 - If the data set sequence number in the JFCB and the data set sequence number in the first HDR1 label on the tape are both greater than 1, the open routine sets the data set sequence number in the UCB to the value of the number in the first HDR1 label. The data set sequence number in the first HDR1 label may be greater than 1 when the volume is part of a multiple data set/multiple volume aggregate.
 - When the processing method is INPUT, OUTPUT, or to the start of a data set on a multiple file tape, the open routine starts with the first value, unless a volume sequence number is specified. If the volume ends with EOVL labels before the desired file sequence number, the open routine switches to the next volume and permanently updates the volume sequence number so that the next open routine to this data set starts with the correct volume.
 - When the processing method is RDBACK, the open routine speeds up the search for the end of the data set by starting with the last volume specified.

If the data set is not yet present on the last volume specified, and if the file sequence number is 1, the open routine can recover by backing up volumes. It detects that the data set is not present if the *dsname* is invalid, the tape starts at a file sequence number greater than 1, or the VOL label is followed by a tapemark.

3. The data set sequence number in the UCB is compared to the requested data set sequence number in the JFCB. If they are equal, the tape is already positioned at the requested data set. If they are not equal, the open routine adjusts the data set sequence number in the UCB as the tape is positioned past each data set, until the number in the UCB equals the number in the JFCB.
4. When multiple tape units are used, and a volume switch causes processing to be continued on a volume on a different unit, the EOVL routine copies the data set sequence number from the previous UCB to the current UCB.
5. If the data set is not open or has been closed, the UCBFSCT field of the UCB is set to X'0000' if:
 - The data set was never opened
 - CLOSE (,REWIND) was specified
 - CLOSE (,REREAD) and LABEL=1 was specified
 - CLOSE (,DISP) was specified or defaulted, and DISP=(,PASS) was not specified on the JCL.

Otherwise, the UCBFSCT field of the UCB has a value that is one greater than the value specified on the LABEL parameter of the JCL.

6. If the job abends while a tape data set is open, the data set will be closed and the tape is positioned as when CLOSE (,LEAVE) is specified. That is, the UCBFSCT field of the UCB has a value that is one greater than that specified on the LABEL= parameter of the JCL.

There are several instances in which a volume is repositioned to the next (or previous) tapemark during open. This is usually done by reading data but suppressing data transmission to storage until a tapemark is found, but can be done by I/O

spacing commands (for example, BACKSPACE FILE). To reduce the chance of an “unexpected record” condition (613-0C), the first method is preferred over the spacing commands. In the event of a 613-08 or 613-0C abend, a tape positioning installation exit (IFG0199I) is given control to allow recovery. For more information about the “data management abend installation” exit, see *DFSMS/MVS Installation Exits*.

Only one data set on a tape volume may be open at any given time. Abnormal termination occurs when you attempt to process a second data set on the same volume.

When the tape is positioned to the data set header label group of the first data set, or the requested data set, the system checks the label identification. Processing is abnormally terminated if the identifier HDR1 is not found.

Processing the HDR1 Label

Verifying Data Set Name

To ensure that the correct data set is being opened, the system compares the requested data set name in the HDR1 label to the data set name specified by the user in the DD statement. This comparison is made on only the 17 least significant (rightmost) characters of the data set name (including 8 characters for the generation and version numbers if the data set is part of a generation data group). DFSMSrmm checks the full 44-character data set name if the previous data set name information is known to DFSMSrmm.

The comparison is made on only the last (rightmost) 17 nonblank characters of the data set name shown in the HDR1 label. It is a good practice, therefore, to limit tape data set names to 17 characters or fewer, when unique names are required, as for password-protected data sets.

For Version 1 HDR1 labels, the generation and version numbers of a generation data set are included in the file identifier and thus are included in the characters compared by the system; however, for Version 3 or Version 4 HDR1 labels, they are not included in the file identifier. For generation data sets with Version 3 or Version 4 labels, the generation and version numbers do not participate in password protection.

For tapes with Version 3 or Version 4 labels, during positioning, the file identifier in each HDR1 label encountered is also checked against duplicate names for the requested data set. The duplicate name check occurs only from the volume position at the time of OPEN until the destination position. For example, if a volume is positioned at the end of data set X at position 5 on the volume as a result of a previous CLOSE LEAVE request, and you request that a data set Y be opened at position 6, no duplicate name check will occur because the volume is already at the desired position. You can force a subsequent duplicate name check from the beginning of the volume by using the CLOSE REWIND option.

If a request is for the first data set on a volume, the volume is always rewound to load point, and no duplicate name search occurs.

You can force a duplicate name check from the first volume of a multivolume configuration by specifying a volume sequence number of 1 (VOL=(PRIVATE,,1) along with a cataloged data set request *or* along with serial numbers specified. During the

search, the message IEC140I START OF DATASET NOT ON VOLUME occurs for each volume mounted on which the requested data set is not found.

Because the suffix of a generation data set is not included in the file identifier of a Version 3 or Version 4 HDR1 label, duplicate name checking prohibits maintaining members of the same generation data group on a volume unless label validation has been suppressed.

If a duplicate name is encountered for a Version 3 label, the label validation exit is entered.

Note: If you inadvertently specify an incorrect file sequence number for a data set, and the data set is encountered before the number specified, the exit is entered with a “duplicate name error” condition. If the exit accepts the error, an abend can result when the target data set does not match the requested data set name.

Processing Label 1 Accessibility Code

For tapes with Version 3 or Version 4 labels, the accessibility field in the HDR1 label is also checked if the volume is not RACF protected. If it contains a valid Version 3 or Version 4 character, the file access exit is entered. For more details about accessibility, see “Protecting Data” on page 94.

Processing Expiration Date

The expiration date shown in the HDR1 label is not verified for input data sets, unless the processing method is INOUT. For INOUT, if the expiration date has not been reached, the system notifies the operator and asks for confirmation of the use of the tape. If confirmation is not received, processing is abnormally terminated. If you override the INOUT specification by coding LABEL=(,IN) on the DD statement, the expiration date is not verified. Automatic handling of expiration date protected tapes is provided by DFSMSrmm functions.

Block Count

The block count shown in the HDR1 label is always an ASCII 0. This 0 is recorded in the DCB and increased during processing for comparison to the block count shown in the trailer label (EOV1 or EOF1).

For reading backward, the block count shown in the trailer label (EOV1 or EOF1) is recorded in the DCB and decreased during processing for comparison to the 0 block count in the HDR1 label.

The block count is verified at the end of the data set or the end of volume. For more information, see “Verifying Block Count” on page 11 and “Verifying Block Count in Label When Reading” on page 109.

Processing the HDR2 Label

Because the HDR2 label is optional with ISCI/ASCII interchange tapes, and the HDR2 format may vary if produced by systems other than MVS, the manner of processing HDR2 labels depends on whether the HDR1 label specifies “IBMZLA” as the name of the system producing the tape. If “IBMZLA” is specified for Version 3, the HDR2 label “Reserved for Operating System” field contains the same data as an IBM standard HDR2 label, and is processed accordingly. See “Processing the HDR2 Label” on page 48 for more information. For Version 4 “Reserved for Operating System” field also contains the CCSID that the data set was written in. If

“IBMZLA” is not specified, the data in the HDR2 label “Reserved for Operating System” and “Record Format” fields cannot be used by the operating system because the contents are unknown.

Positioning After Header Label 2

Unless user header labels are to be processed, the system reads forward until a tapemark is found. All labels that follow the HDR2 label are checked for proper sequence until the tape is positioned at the first data set record (unless validation checking has been suppressed). This includes the optional HDR3-HDR9 labels.

Note: If the HDR2 label is missing, processing will continue with the assumption that it is not needed. If the JFCB/DCB merge function needs it because of missing data set characteristics, the OPEN will abnormally terminate.

Processing User Header Labels

To make the user header labels available to your program, AUL must be coded on the DD statement and the address of an input user header label routine must be specified in the EXLST. The EXLST is described in *DFSMS/MVS Using Data Sets*. If you omit one of these parameters, the system positions the tape past the tapemark immediately after processing the HDR2 label, or the HDR1 label if the HDR2 label is not present.

Read Backward

For the RDBACK processing method, the system uses the data set's trailer labels as header labels, and vice versa. Each label group is read in the normal sequence, that is, EOF1 before EOF2, and so forth. The data records, however, are read in reverse sequence.

Multivolume data sets can be read backward. Concatenated data sets, Format-D (variable-length) records, and Format-S (spanned-length) records cannot be read backward.

End-of-Data or End-of-Volume on Input

The system's EOVR routine handles both end-of-data-set and end-of-volume conditions on input. These conditions occur when either a tapemark is read or an EOVR macro instruction is executed by the processing program.

After encountering a tapemark, the system checks the first 4 bytes of the first trailer label for the identifier EOVR1 or EOF1. Processing is abnormally terminated if neither identifier is found.

For an end-of-data condition, the first trailer label is processed, unless deferred user input trailer label processing is specified in the DCB exit list. For an end-of-volume condition, the first trailer label on the current volume is processed, and then the volume label and header labels on the next volume are processed.

Except when it is used as a header label for a RDBACK operation, the system ignores the second trailer label (EOVR2 or EOF2) of an input data set.

Unless the tape is read backward and the trailer labels are used as header labels, trailer labels are not validated for Version 3 or Version 4 standards.

When the FEOV macro is issued, the identifier of the first trailer label is not checked. The trailer labels on the current volume are not processed, but the volume labels and header labels on the next volume are processed.

If UTLs are present on input, the system can make them available to your program. To make them available, AUL must be coded on the DD statement and the address of an input user trailer label routine must be specified in the DCB exit list.

Verifying Block Count in Label When Reading

To verify that all records on the input data set on the current volume have been read, the system compares the block count shown in the first trailer label (EOV1 or EOF1) against the block count that was accumulated in the DCB. For reading backward, the system compares the 0 block count shown in the HDR1 label against the block count in the DCB.

If the block count in the label does not equal the block count in the DCB, the EOV routine gives control to the appropriate entry in the user's DCB exit list. This entry in the exit list is identified by the code hex 0B. The EOV routine passes the following information to the exit routine:

Register	Contents
0	The block count shown in the label
1	The address of the DCB.

After your exit routine analyzes the discrepancy (and possibly prints a message), your exit routine must return to the EOV routine with one of the following return codes in register 15:

ReturnCode	Meaning
0	Abnormally terminate with completion code of X'237'
4	Continue processing.

For more information, see *DFSMS/MVS Using Data Sets* and “Verifying Block Count” on page 11.

If the appropriate user exit entry is not provided in the DCB exit list, or a return code of 0 is received, then a block count discrepancy causes the system to call the optional DCB ABEND exit and issue ABEND 237. When the FEOV macro is executed, the block count is not verified. For more information, see *DFSMS/MVS Using Data Sets*.

Determining Volume Switch

For a multivolume input data set, you must specify the serial numbers of all the volumes to be processed. Specify the serial numbers either directly in the DD statement or indirectly through the catalog procedure. You specify the serial numbers in forward sequence, regardless of whether the tapes are to be read forward or backward.

- For noncataloged data sets, you specify the volume serial numbers in the VOLUME parameter of the DD statement. The system processes the group of volumes in whatever order you specify and processes only the volumes you specify.
- For cataloged data sets, process the group of volumes in sequential order.

However, you can begin processing at any volume of the group by specifying a volume sequence number in the VOLUME parameter of the DD statement.

For input, the label identifier of the trailer labels determines whether the system continues processing the data set. When the system finds an EOV label, it performs volume switching. When the system finds an EOF label, it passes control to the user's end-of-data routine, with these exceptions:

- If the DD statement specifies OPTCD=B, and if an additional volume is available, then the system performs volume switching. If OPTCD=B, no volumes are available for switching, and no data set concatenated after the current one, then the system passes control to the user's end-of-data routine.
- If another data set is concatenated after this one, the system processes it. If the user has not set on the unlike-attributes bit (the DCBOFPPC bit), then the system treats the concatenated data set as if it began another volume of the current data set. If the user defined an EOV exit, the system calls it (not a DCB open exit). If the unlike-attributes bit is on, then the system closes the current data set and opens the concatenated data set. If the user defined a DCB open exit, the system calls it (not an EOV exit). For more information on using concatenated data sets, see *DFSMS/MVS Using Data Sets*.

To determine whether additional volumes are required, the system maintains a volume sequence number in the DEB in storage.

- For read forward operations, the volume sequence number in the DEB increases as each volume is processed. This count is compared to the total number of volumes requested, as shown in the JFCB.
- For RDBACK operations, the volume sequence number in the DEB is initialized to the total number of volumes requested, as shown in the JFCB. The DEB count is decreased as each volume is processed, until the count reaches 0.

If another volume is not required (end-of-data-set condition), control is given to the user's end-of-data-set routine that is specified in the DCB. Subsequently, the processing program or the operating system closes the data set. The user's end-of-data-set routine is not entered until the last specified volume or the last concatenated data set is processed. If an input data set is closed before the end of the data set is reached, the user's end-of-data-set routine is not entered.

If another volume is required (end-of-volume condition), the system obtains the next volume serial number from the JFCB and performs volume switching. If the new volume is not already mounted, the EOV routine issues a mount message to the operator.

When multiple tape units are being used, the EOV routine also checks to see if there is a next-plus-one volume specified, and if the volume just completed can be rewound and unloaded. If so, the EOV routine issues a message directing the operator to mount the next-plus-one volume on the tape unit just used. This is a pre-mounting aid; the next-plus-one volume label is not verified at this time. This lookahead or parallel mounting results in an IEC501E mount message.

Note: The IEC501E message has a type 2 descriptor code that causes the message to be retained on the screen.

If you do not want message retention, turn off the message retention attribute. You can accomplish this by either:

- Specifying RETAIN(NO) for the message id in an MPFLSTxx record
- Using a WTO message exit (an MPF specific exit or IEAVMXIT) to alter the message descriptor code or to request “do not retain this message.”

For tape mounts, the IEC501E message remains on the screen until the mount is satisfied or until EOV has occurred on the previous volume. EOV causes a DOM to be initiated against the IEC501E message and issues the IEC501A message.

Checking the Next Volume

When volume switching is performed for multiple volume input, the EOV routine checks the volume and header labels on the new volume.

The volume serial number of the volume label, VOL1 label, and data set label are checked to verify that the correct volume is mounted.

The method of locating and checking the HDR1 label varies according to the situation. The processing depends on whether the data set is a continuation of a multi-volume data set or is a concatenated data set with like characteristics. Data sets with like characteristics can be processed correctly using the same DCB, input/output block (IOB), and channel program.

- *Multivolume data set:* The data set sequence number is irrelevant for the second and subsequent volumes of a multivolume data set. The EOV routine assumes that the data set continues at the beginning of the new volume and, therefore, checks the first header label group on the tape. The HDR1 label is checked in the same manner as when the data set was opened on the first volume.
- *Concatenated data sets:* The EOV routine handles concatenated data sets with like characteristics. Such data sets are not necessarily the first on the volume, so the EOV routine positions the tape according to the specified data set sequence number. This positioning is the same as for opening a data set. The HDR1 label is checked as it was when the first data set was opened.

The HDR2 label on the new volume is not processed. The data set characteristics that were established when the data set was opened apply to all subsequent volumes handled by the EOV routine. However, the HDR2 label is validated for Version 3 standards.

The data set's block count is not accumulated from volume to volume. It is initialized and verified separately for each volume.

Closing an Input Data Set

The close routine does not process trailer labels on an input data set. Usually, the trailer labels are processed by the EOV routine before the data set is closed unless deferred user input trailer label processing was specified in the DCB exit list. If deferred user input trailer label processing was specified, the processing otherwise performed for an input end-of-data condition is performed when an input data set is closed.

If an input data set is closed before it reaches the end of data set or the end of volume, or if the FEOV macro is executed, processing of trailer labels is not performed.

Opening an Output Data Set

When a data set is opened for output, processing is similar to opening for input, except that:

- Only Version 3 or Version 4 tapes are created for output processing.
- If the system-wide RACF tape protection option has been specified, and the DD statement has specified PROTECT=YES but has not previously been opened for output processing, OPEN ensures that the PROTECT=YES specification is valid. Both the volume sequence number and the file sequence number must be set to 1 and a private volume must be requested. The protection indicator in the JFCB is reset so that subsequent OPENS of that DD statement for output processing do not attempt validity checking (of the PROTECT=YES specification) and definition of the volume to RACF.
- The label anomaly exit and volume label editor are entered for label conflicts. The volume label editor is also entered for version conflicts (the label is not Version 3 or Version 4) if output is to the first data set. If output is to any data set other than the first data set, or if the first data set is allowed to be extended, a version compatibility conflict causes the volume to be rejected.
- An action message is issued to the operator if the tape is file protected (to allow writing).
- If the system-wide RACF tape protection option has been specified, RACF authorization at the UPDATE level is checked. If a Version 3 or Version 4 tape is not RACF protected, the volume accessibility code is checked as it is for input processing. For additional information, see “Protecting Data” on page 94.
- Symmetry violations during output to a Version 3 volume occur if the open option is EXTEND or OUTINX. Open for MOD during output also violates symmetry, and is checked after the volume has been positioned. An EXCP DCB is checked to ensure the presence of a device-dependent area large enough to contain a block count.

If a density conflict occurs, the volume label editor is entered. If the conflict occurs for the first data set on the volume, a new volume label (Version 3 or Version 4) is written in the density specified by the user. For other than the first data set, the volume label is written in the density of the volume label at the beginning of the tape, unless the volume label editor is modified. If the old volume label is longer than 80 characters, the excess characters are lost unless:

- The IBM-supplied volume label editor is replaced with a program that protects the extra data during a density conflict before returning to open/EOV for reversionification of the volume, or
- The operator rejects a rewrite of the label by a response to message IEC704A L or IEC704 L UVL.

For IDRC compaction capable tape drives, the system enforces single mode recording. This results in the second and subsequent data sets on the volume being written in the same mode as the first data set. Since the existing mode is

determined by information in the volume label, single mode recording is not enforced for NL volumes.

New header labels are written after the volume label is rewritten.

Attention Doing multiple opens and closes without writing any user data in the end-of-tape reflective marker area can result in the writing of header and trailer labels past the marker. Access methods detect the reflective markers; however, since the creation of empty data sets does not involve access methods, the end-of-tape marker is not detected. This can cause the reel to run out of tape.

Volume Serial Number

You are not required to specify the volume serial number for output tapes, unless you are extending an existing tape data set. Serial numbers are required when the processing method is EXTEND or OUTINX or OUTPUT DISP=MOD. If no number is specified, the mount message directs the operator to mount a scratch tape. The system obtains the volume serial number from the VOL1 label and records it in the JFCB and the UCB.

If you choose to specify the volume serial number, the system compares it with the volume serial number shown in the VOL1 label. If the serial numbers match, then the system resets a mount switch in the UCB to indicate that volume mounted is verified. The mount switch is initially set when the mount message is issued to the operator. If the serial numbers do not match, the system may give the operator the option of having the label rewritten with the serial number of the volume requested. This will occur if the tape is not password protected, is not date protected, and is not a checkpoint/restart volume. Otherwise, the system rejects the tape and issues another mount message.

Positioning the Volume to the Data Set

When the volume is mounted and verified, the system positions the tape to receive the new data set. Usually, the new data set is the first and only data set on the tape, so the tape remains positioned immediately following the VOL1 label.

To create a data set that follows another data set already stored on the tape, you specify a data set sequence number in the LABEL parameter of the DD statement.

- The sequence number can be from 1 to 9999, with 1 representing the first data set on the volume. If the volume ends with EOVS labels before the specified sequence number, the open routine will switch to the next volume. If you specify a sequence number that is greater than the number of data sets existing on the volume, plus one, your task will be abnormally terminated. For any label validation errors encountered beyond the 254th data set, the error message does not include the explicit sequence number; instead, it indicates "254+."
- If you do not specify a sequence number, or if you specify 0, the system assumes that the data set is to be written as the first one on the volume.

To position the tape, the system maintains a logical data set sequence number in the UCB. The method of positioning is the same as that previously explained for opening an input data set.

Only one data set on a tape volume can be open at any time. Processing is abnormally terminated if you attempt to open another data set on the same volume.

When the tape is positioned to receive the new data set, the system expects to find either an existing HDR1 label or a tapemark. If neither is present, the system assumes that other data is recorded where the HDR1 label should be and, therefore, processing is abnormally terminated. If the last data set on a tape has EOVL labels, another data set cannot be written to follow it.

If a tapemark is found, it indicates that a HDR1 label does not exist at the position at which the new data set is to be written. The system bypasses all further label verification and accepts the tape for output. The conditions under which the system finds a tapemark instead of a HDR1 label are:

- When a tapemark immediately follows the VOL1 label. This can occur when the tape is initialized by means other than the EDGINERS or IEHINITT utility programs (both write a dummy HDR1 label following the VOL1 label). The tapemark is overwritten by the new HDR1 label.
- When, for multiple data set organizations, the new data set is to be written after the last existing data set on the volume. In this case, the system encounters the second tapemark following the existing EOF trailer label group. The tapemark is overwritten by the new HDR1 label.

Duplicate data set names are checked for Version 3 volumes during positioning, as described under “Opening an Input Data Set” on page 103. If a volume is not RACF protected, the accessibility code in the existing HDR1 label of a Version 3 or Version 4 volume is checked before it is overwritten with a new HDR1 label.

Expiration Date on Existing Label

For a Version 3 volume with multiple data sets, the system checks the expiration date of the data set that immediately precedes the output data set. If the expiration date of the previous data set is later than the expiration date of the output data set, unless label validation has been suppressed, the label validation exit is entered.

If the previous data set's expiration date is equal to or greater than that of the output data set, then the expiration date of the output data set is compared with the current date. If the expiration date of the output data set has not been reached, then the operator confirms the use of the tape or mounts another tape. DFSMSrmm provides facilities that enable automation of handling expiration protection. See *DFSMS/MVS DFSMSrmm Implementation and Customization Guide* for more information.

Any data sets following the output data set are treated as if they expired on the same day as the output data set.

Writing Data Set Header Labels

When the tape is accepted by the system for output, the system creates the header labels, HDR1 and HDR2, for the new data set. These labels are created from information in the updated JFCB and other system control blocks.

The source of information for each field of a label is explained in the description of label formats. The process of updating the JFCB is explained in Figure 2 on page 4.

If no user header labels are written, then the system writes a tapemark after the HDR2 label. The tape is then ready to receive the new data set.

Writing User Header Labels

For the system to write UHLs, you must code AUL on the DD statement and specify the address of an output user header label routine in the EXLST. The EXLST is described in *DFSMS/MVS Using Data Sets*.

Unless label validation has been suppressed, the user header labels are validated to ensure they contain valid ASCII characters. Violations cause the label validation exit to be entered.

User header labels do not have to be sequentially numbered or lettered. There is no limit to the number that can be associated with a data set.

Permanent I/O Error

In some cases, if a permanent I/O error occurs during label processing, and the data set is the first one on the first or only volume, then the operator removes the tape and mounts a scratch tape, even if you request a specific volume. The job will be abnormally terminated if the data set is not the first one on the volume or is not the first volume of a multivolume data set.

EOV on Output

The system's EOV routine automatically switches volumes when one of the following EOV conditions occur:

- The reflective strip on a reel is encountered
- The logical end of a tape cartridge is reached
- The FEOV macro is executed.

This volume switching includes:

1. Checking whether the expected number of blocks were read or written (see "Verifying Block Count" on page 11)
2. Writing trailer labels on the current volume
3. Checking existing volume and header labels on the new volume
4. Writing a volume label on the new volume (in the case of a density conflict or version conflict)
5. Writing header labels on the new volume.

If a density or a version conflict occurs when the volume label is checked, a new label (Version 3 or Version 4) is written in the density or the version specified by the user. If any VOL2–VOL9 or user volume labels (UVLs) exist, they are overwritten. If the original volume label is longer than 80 characters, the excess characters are truncated when the new volume label is written.

When multiple tape units are used, the EOV routine checks to see if a next-plus-one volume is needed, and if the volume just written can be rewound and unloaded. If so, the EOV routine issues a message directing the operator to mount the next-plus-one volume on the tape unit just used. This is a premounting aid; the next-plus-one volume label is not verified at this time. This lookahead or parallel mounting results in an IEC501E mount message.

Note: The IEC501E message has a type 2 descriptor code that causes the message to be retained on the screen.

If you do not want message retention, turn off the message retention attribute. You can accomplish this by either:

- Specifying RETAIN(NO) for the message id in an MPFLSTxx record
- Using a WTO message exit (an MPF specific exit or IEAVMXIT) to alter the message descriptor code or to request “do not retain this message.”

For tape mounts, the IEC501E message remains on the screen until the mount is satisfied or until EOV has occurred on the previous volume. EOV deletes the IEC501E message and issues the IEC501A message.

Writing Data Set Trailer Labels

Trailer labels are always written at an EOV condition on output tapes and are identified as EOVS1 and EOVS2 (as opposed to EOF for end of data). These labels are created in the same manner and with the same content as the data set header labels, except for the label identifiers and the block count.

Two tapemarks are written at EOV following the data set trailer labels. If user trailer labels are written, the tapemarks follow the user labels.

Writing User Trailer Labels

When AUL is coded on the DD statement and the address of an output user trailer label routine is specified in the EXLST, the system can write as many user trailer labels as desired.

The contents of UTLs are not checked during EOV processing, but, if they contain any invalid ISO/ANSI characters, the condition is detected during a subsequent open for RDBACK.

Labels on New Volume

The EOV routine handles label processing on the new volume (checking existing labels and writing new labels). The processing is the same as the open routine's handling of the first volume, except the version level of the new volume will be the same as the version level of the previous volume. It will not be allowed to create a mixed version multi-volume data set. All volumes in a multi-volume data set must be the same version, either Version 3 or Version 4.

When you create a multivolume data set, the data set sequence number is irrelevant for the second and subsequent volumes. The EOV routine assumes that the data set continues at the beginning of the new volume.

RACF Processing on the New Volume

If the system-wide RACF tape protection option has been specified, RACF authorization checking occurs for the new volume. If the previous volume is RACF protected, the new volume must either be not defined to RACF (in which case, EOV will define it to RACF as part of the previous volume's volume set profile), or the new volume must be defined as part of the same volume set profile as the previous volume. If the previous volume is not RACF protected, then the new volume is

rejected if a nonspecific request is made, or the program will be abnormally terminated if a specific request is made.

For tapes with Version 3 or Version 4 labels, the volume access exit is entered if the new volume is not RACF protected and the accessibility field in the VOL1 label contains a valid Version 3 or Version 4 character.

Special EOV Conditions

The following conditions signal the end of the volume:

- The drive signals to the system that it is nearing the end of the tape.
 - On a reel, the system detects the end-of-tape reflective marker.
 - On a cartridge, internal mechanisms sense the end of the tape.
- An FEOV command is issued.

The EOV routine writes the trailer labels as described above. If the drive signal occurs while the system is writing the trailer labels, the EOV or the close routine continues to write the trailer labels. In both cases, the data set can be read or over-written normally.

If the system is writing header labels, and the drive signals that it is nearing the end of the tape, the system:

- Writes the new header label group
- Allows the user to write one record
- Writes the new trailer label group
- Performs volume switching.

Closing an Output Data Set

The close routine handles end-of-data-set processing on output tapes. When a write operation is the last operation that occurs before closing a data set (for OUTPUT, OUTIN, OUTINX, EXTEND, or INOUT), or when no output is written before closing (for OUTPUT, OUTIN, OUTINX, EXTEND, or INOUT), the close routine creates data set trailer labels.

Writing Data Set Trailer Labels

The close routine writes the data set trailer labels with the identifiers EOF1 and EOF2. Except for the label identifiers and the block count, these labels are created in the same manner and with the same content as the data set header labels.

The close routine writes two tapemarks following the trailer labels. If user labels are to be written, the tapemarks follow the user trailer labels. If another data set is added to the tape (multiple data set organization), then that data set's HDR1 label overlays the second tapemark.

Writing User Trailer Labels

When AUL is coded on the DD statement and the address of an output UTL routine is specified in the DCB exit list, the close routine can write as many UTLs as desired.

The contents of the UTL are not checked during close processing, but if they contain any invalid ASCII characters, the condition is detected during a subsequent open for RDBACK.

Checkpoint/Restart Not Allowed

Any ISO/ANSI tape that is open during a CHKPT macro service prevents a checkpoint from being taken.

Chapter 4. Nonstandard Labels

Nonstandard labels do not conform to the IBM or ISO/ANSI standard label formats. They are labels that you design. Your installation also must provide routines to write and process them. There are no requirements for the length, format, content and number of nonstandard labels, except that the first record on a BCDIC, EBCDIC, or ASCII tape cannot be a standard volume label.

DFSMSrmm does not support nonstandard labeled tapes.

Nonstandard label routines are inserted into the control program by link-editing them into SYS1.LPALIB. Only an authorized system programmer can update SYS1.LPALIB. IBM recommends that you use SMP/E to link-edit.

Nonstandard Label Processing Routine	Control Program Routine	Member Name
Input Header	Open	NSLOHDRI
	EOV	NSLEHDRI
Output Header	Open	NSLOHDRO
	EOV	NSLEHDRO
Input Trailer	EOV	NSLETRLI
Output Trailer	EOV	NSLETRL0
	Close	NSLCTRLO
Restart	Restart	NSLRHDRI

If you use nonstandard tape labels and you want to use the DDR option, your system program must provide an exit routine to perform volume verification. The routine must be able to perform your verification within the first 48 bytes of any record in your nonstandard label. An LPA routine named NSLREPOS must be provided.

For detailed information about these and other available exit routines, see *DFSMS/MVS Installation Exits*.

Nonstandard Volume Label Verification and Editor Routines

If you specify that an input or output tape has a nonstandard label, the operating system checks for the nonstandard volume label at the beginning of the tape.

Because of conflicting label types or conflicting tape characteristics, various error conditions can occur during the verification of the first record. Under some error conditions, the tape is accepted for use. Under other error conditions, the tape is not accepted and the system issues another mount message. For certain other error conditions, the system gives control to a *volume label editor routine*. Your installation can use IBM-supplied routines, or it can supply its own routines.

The IBM-supplied volume label editor routines determine the discrepancies between the requested tape and the mounted tape and, if necessary, pass control to the

appropriate system routine to create or destroy labels. DFSMSrmm supplies a label editor that prevents the creation and destruction of standard labels. Installation-supplied routines can perform other functions.

If your installation supplies its own volume label editor routines, the first (or only) module of each routine must be named as follows:

- OMODVOL1 (for the editor routine associated with open)
- EMODVOL1 (for the editor routine associated with EOVS).

If either of your installation's editor routines consists of more than one load module, the names for the additional modules must begin with the prefix OMODVOL for the open routine or EMODVOL for the EOVS routine. Transfer between the modules must be by name.

For detailed information about these and other available exit routines, see *DFSMS/MVS Installation Exits*.

Verifying Block Count

At the end of the data set on each cartridge (not reel), EOVS or CLOSE routines can check whether the expected number of blocks were read or written. See "Verifying Block Count" on page 11.

Chapter 5. Unlabeled Tapes

To process or create a tape with no labels, specify NL or LTM (for VSE tapes) in the LABEL parameter of the DD statement. An unlabeled tape contains only data records and tapemarks. The organization of data sets on one or more volumes is shown in Figure 22 on page 122. The system automatically writes the tapemarks on output and expects to find a similar placement of tapemarks on input.

- A tapemark does not precede the first data set on any volume if you specify NL. A tapemark can precede the first data set if you specify LTM. If you specify NL for a tape with a tapemark as the first record, it means the first data set contains no data. If you specify LTM for an unlabeled tape that does not contain a leading tape mark, the system treats the request the same as for NL. See “Unlabeled Tapes” on page 130 for more information.
- A tapemark follows each data set.
- Two tapemarks follow a data set if it is the last or only data set on the volume.

An unlabeled tape can be read backward even though there is no tapemark preceding the first data set. In this case, the end-of-data-set condition is signaled by the reflective strip or other indicator at the beginning of the tape.

Open/Close/EOV lookahead mounting does not accept any unlabeled volume that is mounted before the system requests it (UCBNRY=0 and UCBVOLI=zeros). A demount message with a blank *vol ser* is issued.

Note: AVR automatically recognizes mounts of labeled volumes. If JCL-specific requests are used and a volume that is not specified is mounted at allocation time, the system requests a demount of the incorrect volume and a mount of the specified volume. Avoid mounting unlabeled tapes before the system requests them.

To use the BLP option, the user is responsible for proper tape positioning because Open/Close/EOV cannot determine if the tapes are labeled or unlabeled. BLP is identical to NL processing, except that the check for an existing label is bypassed. BLP positions second and subsequent volumes of a multivolume data set as if it were unlabeled. This positioning is incorrect if the data set has labels. If BLP is requested under DFSMSrmm, then DFSMSrmm checks to see if the volume is labeled. If the volume and the label do not match, then the requested volume is rejected.

For more information about BLP see “Bypassing Label Processing” on page 11

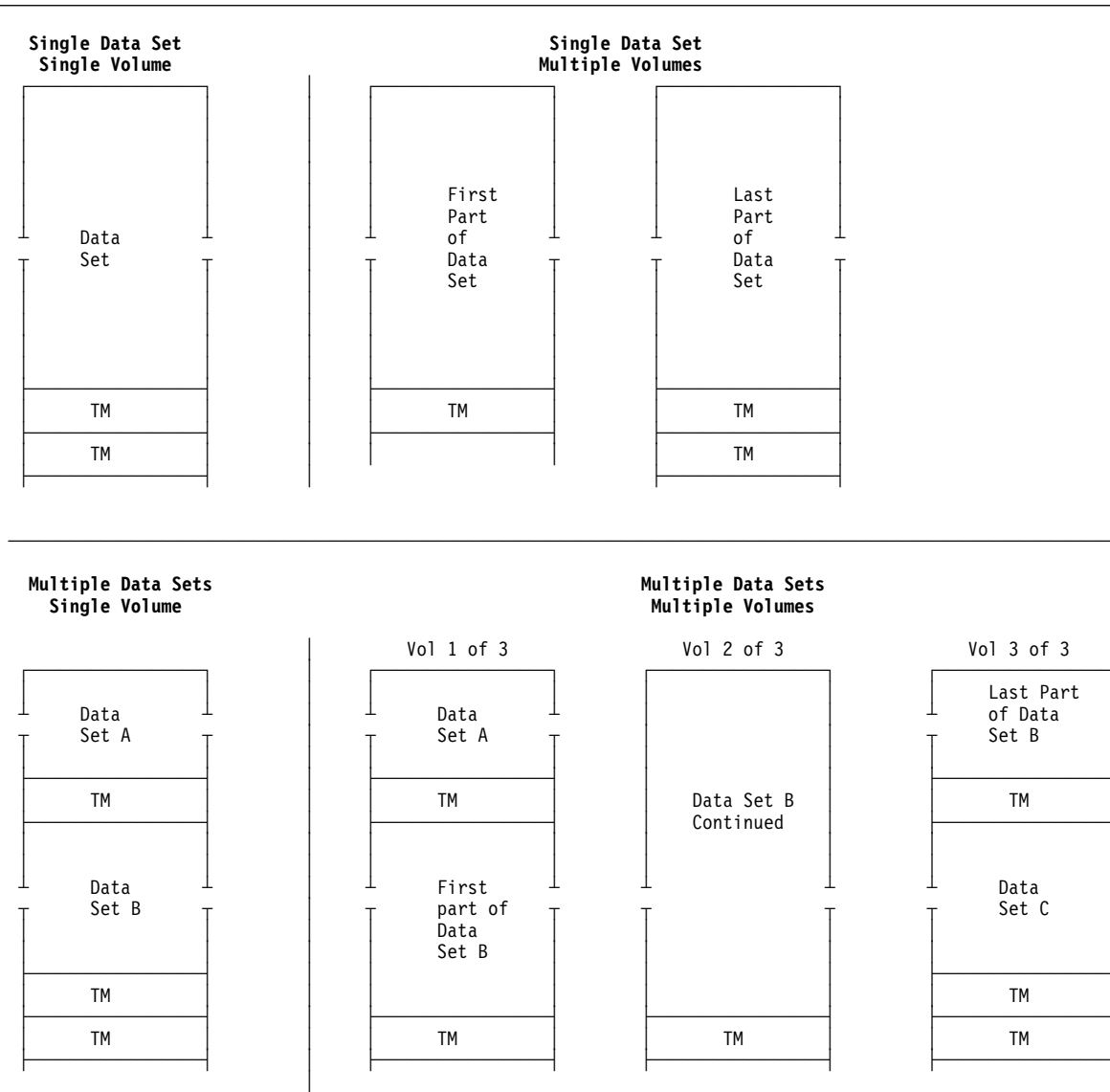


Figure 22. Organization for Unlabeled Tapes

Creating an Unlabeled Tape Volume with DFSMSrmm

You can use the EDGINERS utility to create an unlabeled tape volume. DFSMSrmm verifies that the correct volume is mounted, either by reading any existing label or by asking the operator to confirm the external label. DFSMSrmm writes an 80-byte record that is not recognized as any valid label format, however, the DFSMSrmm utility can recognize it as an unlabeled tape once it is initiated.

Opening an Input Data Set

When you specify no labels, the system checks the input tape to ensure that the first record is not a standard volume label (that is, the first 4 characters are not VOL1 in EBCDIC or ASCII). If the first record is a standard volume label, the tape is rejected with a message from the system directing the operator to mount the correct tape. The various error conditions that can occur during label verification are

explained under “IBM Standard Volume Label Verification and Editor Routines” on page 43, “ISO/ANSI Volume Label Verification and Editor Routines” on page 94, and “ Nonstandard Volume Label Verification and Editor Routines” on page 119.

The search for a standard label is the only mount verification performed by the system. Without labels, neither the volume nor the data set can be positively identified and the system assumes that they are correct. The operator checks the reel's external identification to ensure compliance with the mount message.

Positioning the Volume to the Data Set

When the tape is accepted for input, the system positions the tape at the first record of the data set to be processed. Usually there is only one data set on the volume and positioning is set to the first record on the tape.

To retrieve a data set when there are more than one on a single reel of tape, specify a data set sequence number in the LABEL parameter of the DD statement, unless the data set is cataloged. You need not specify a data set sequence number for a cataloged data set, because the number can be obtained from the catalog along with the volume serial number.

- The sequence number can be from 1 to 9999, with 1 representing the first data set on the volume. If you specify a sequence number higher than the number of data sets on the volume, then the tape is spaced through and removed from the reel.
- If you do not specify a sequence number or a zero, and the data set is not cataloged, then the system assumes that the data set is first in sequence on the volume.
- The first data set on an unlabeled tape is not preceded by a tapemark. If a tapemark precedes the first data set, the sequence number of that data set is two (the effect is as if a data set containing no data preceded the tapemark).

To position the tape, the system uses the requested data set sequence number in the JFCB and maintains a logical data set sequence number in the UCB. The number in the UCB represents the current position of the tape and is maintained as follows:

1. When a tape is first mounted, the data set sequence number in the UCB is 0.
2. When a data set is opened, the open routine sets the data set sequence number in the UCB to 1. If the tape is still positioned from previous processing, such as for a LEAVE request, the open routine does not reset the number in the UCB. This means that multiple volume, multiple data set NL tape, unlike SL tape, requires a different JCL data set sequence number, unless the tape stays mounted.
3. The data set sequence number in the UCB is compared to the requested data set sequence number in the JFCB. If they are equal, the tape is already positioned at the requested data set. If they are not equal, the open routine adjusts the data set sequence number in the UCB as the tape is positioned past each data set until the number in the UCB equals the number in the JFCB.
4. To position to a data set when there are multiple data sets on multiple volumes, specify the volume serial numbers of the volumes on which the data set resides in the VOLUME parameter of the DD statement and the data set sequence

number in the LABEL parameter. You are not required to specify the volume serial number or the data set sequence number for a cataloged data set.

5. When multiple tape units are used and a volume switch causes processing to be continued on a volume on a different unit, the EOV routine copies the data set sequence number from the previous UCB to the current UCB.

No more than one data set on a tape volume can be open at a time. Processing is abnormally terminated if you attempt to begin processing a second data set on the same volume.

Read Backward

For the RDBACK operation, the data records are retrieved in reverse sequence. Multivolume data sets can be read backward. Concatenated data sets cannot be read backward. Format-V (variable-length) records cannot be read backward. Seven-track tape with data conversion cannot be read backward.

End-of-Data or End-of-Volume on Input

For input, the system's EOV routine handles both end-of-data-set and end-of-volume conditions. These conditions occur when either a tapemark is read or an FEOV macro instruction is executed by the processing program. If the tape is a cartridge, the EOV routine can check whether the expected number of blocks were read. See "Verifying Block Count" on page 11.

Determining Volume Switch

The volume serial numbers of the data set to be processed must be specified by the user at execution time. The serial numbers are specified either directly in the DD statement or indirectly through the catalog procedure. You specify the serial numbers in forward sequence, regardless of whether the tapes are to be read forward or backward.

- For noncataloged data sets, you specify the volume serial numbers in the VOLUME parameter of the DD statement. The system processes the group of volumes in whatever order you specify and processes only the volumes you specify.
- For cataloged data sets, the group of volumes must be processed in sequential order. However, you can begin processing at any volume of the group by specifying a sequence number in the VOLUME parameter of the DD statement.
- For input, the volume serial numbers specified by the user are the basis for determining whether a volume switch is required. The system does not consider whether the data set on the current volume is followed by one or two tapemarks. To determine whether additional volumes are required, the system maintains a volume sequence number in the DEB in storage.
- For read forward operations, the volume sequence number in the DEB is increased as each volume is processed. This count is compared to the total number of volumes requested, as shown in the JFCB.
- For read backward operations, the volume sequence number in the DEB is set to the number of volumes requested, as shown in the JFCB. This count in the DEB is decreased as each volume is processed until the count equals 0.

If another volume is required, the system obtains the next volume serial number from the JFCB and switches volumes. The system checks the initial record of the new volume to ensure that it is not a standard volume label and positions the tape to the data set. For a multivolume data set, the tape is positioned to the first record on the new volume. For a concatenated data set, the tape is positioned according to the specified data set sequence number.

If another volume is not required, control is given to the user's end-of-data-set routine that is specified in the data control block. Subsequently, the processing program or the operating system closes the data set.

- The user's end-of-data-set routine is not entered until the last specified volume, or the last concatenated data set is processed.
- If an input data set is closed before it reaches the end of the data set, the user's end-of-data-set routine is not entered.

Opening an Output Data Set

When you specify unlabeled tape, the system checks the output tape to ensure that the existing first record is not a standard volume label. If the first record is 80 bytes in length and contains the identifier VOL1 in the first 4 bytes, the system checks for the following conditions in the order presented:

1. RACF authorization
2. Password protection
3. Expiration date.

If the system-wide RACF tape protection option has been selected, the system checks the alter level authorization to the tape volume. If the tape volume is not defined to RACF, then password protection is checked. If the tape volume is defined and the user is not authorized for ALTER, then the tape is demounted. If the user is authorized for ALTER, then password protection is bypassed.

If the system determines that the volume is password protected, then a message to demount the tape is issued to the operator. Otherwise, the system continues processing by checking the expiration date. If the expiration date is earlier than the current date, then a message is issued, and the operator can either refuse or allow the use of the tape. If the expiration date is later than the current date, then another message is issued, and again the operator can refuse or allow the use of the tape. If in either situation the operator refuses the use of the tape, then the system requests that another volume be mounted. If the operator accepts the tape, then the system destroys the standard label by writing a tapemark over it, thus providing you with the unlabeled tape you requested.

DFSMSrmm can prevent standard labelled tapes from being converted into unlabeled tapes.

If the tape volume has been found to be RACF defined and the user is authorized for ALTER, then that definition is deleted when the label is destroyed. If you do not want the system to perform this checking for you, then you should insert your own label editor routine.

For more information on label editor routines and the various error conditions that can occur during verification of the first record, see "IBM Standard Volume Label

Verification and Editor Routines” on page 43, “ISO/ANSI Volume Label Verification and Editor Routines” on page 94, and “Nonstandard Volume Label Verification and Editor Routines” on page 119.

Attention Doing multiple opens and closes without writing any user data past the end-of-tape marker can result in the writing of header and trailer labels past the marker. Access methods detect the markers; however, since the creation of empty data sets does not involve access methods, the end-of-tape marker is not detected. This can cause the reel or cartridge to run out of tape.

Volume Serial Number

You are not required to specify volume serial numbers for unlabeled output tapes. If none is specified, the mount message directs the operator to mount a scratch tape.

DFSMSrmm prevents the use of unlabeled scratch tapes because there is no way to tell which volume is mounted or to manage them without operator intervention.

If you request a specific volume, the operating system uses the specified volume serial number for mounting messages, for cataloging, and for passing the volumes to other job steps.

If you do not request a specific volume, the system cannot obtain the actual serial number of the volume that is mounted. In this case, the system generates a volume serial number and assigns it to the volume. These volume serial numbers are generated in the form Lxxxxyy, where:

- xxx is a number the open routine increments (by one) each time an output data set is opened on a nonspecified unlabeled volume. If more than one data set is created on the same volume, this number is increased only when the first data set is opened.
- yy is set to 00 by the open routine. The EOVS routine increments this number (by one) each time an end-of-volume condition occurs. In this way, each volume of a multivolume data set is assigned a different volume serial number.

If a data set is to be cataloged, you should specify the volume serial numbers for all the volumes required. This prevents different data sets residing on different volumes from being cataloged with identical volume serial numbers, which could result in the mounting of wrong volumes.

Positioning the Volume to the Data Set

When the tape is accepted for output, it is positioned to receive the new data set. Usually, the new data set is the first or only data set on the volume, so the tape is positioned at load point.

To create a data set that follows another data set already stored on the volume, specify a data set sequence number in the LABEL parameter of the DD statement.

- The sequence number can be from 1 to 9999 with 1 representing the first data set on the volume. If you specify a sequence number that is 2 or more greater than the number of data sets existing on the volume, one of two things may happen:
 - the tape will be spaced through and removed from its reel, or

- the data set will be written but separated from the preceding data set by unusable (old) data.
- If you do not specify a sequence number, or if you specify 0, the system assumes that the data set is to be written as the first one on the volume.

To position the tape, the system maintains a logical data set sequence number in the UCB. The method of positioning is the same as that previously explained for opening an input data set.

No more than one data set on a tape volume can be open at a time. Processing is abnormally terminated if you attempt to open a second data set on the same volume.

End-of-Volume on Output

The system's EOVR routine automatically switches volumes when one of the following EOVR conditions occur:

- The reflective strip on a reel is encountered
- The logical end of a tape cartridge is reached
- The EOVR macro is executed

The EOVR routine:

- Checks whether the expected number of blocks were written (see "Verifying Block Count" on page 11)
- Writes one tapemark after the data set on the current volume
- Checks the new volume to ensure that the volume does not contain a standard volume label

The output is then continued on the new volume.

The EOVR routine writes one tapemark after the data set on the current volume and checks the new volume to ensure that the volume does not contain a standard volume label. The output is then continued on the new volume.

Closing an Output Data Set

The close routine handles end-of-data-set processing on output tapes. When a write operation is the last operation that occurs before closing a data set (for OUTPUT, OUTIN, OUTINX, or INOUT) or when no output is written before closing (for OUTPUT, OUTIN, or OUTINX), the close routine creates data set trailer labels.

Restarting from a Checkpoint

When a job step is restarted from a checkpoint, the restart routine mounts and positions any tape volumes containing data sets that were open when the checkpoint was taken. Specifically, the restart routine:

- Restores applicable control blocks to the conditions that existed when the checkpoint was taken.
- Ensures that the first existing record on the tape is not a standard volume label (VOL1).

- Uses the data set sequence number shown in the JFCB to position the tape at the required data set. The method of positioning is the same as previously explained for opening an input data set.
- Uses the block count shown in the DCB to reposition the tape at the proper record within the data set. For forward read operations, this positioning is performed in a forward direction. If the block count is 0 or negative, the tape remains positioned at the interrecord gap preceding the first record. For backward read operations, this positioning is performed in a backward direction. If the block count is 0 or a positive number, the tape is positioned at the interrecord gap following the last record of the data set.

Chapter 6. Using Tape Volumes Created by Other Systems

Occasionally, it is necessary to process a tape volume created by another system. There is no exact procedure: many of the factors vary according to the situation and the user's options at the time the volume was created. The volume can be slightly or extremely different in its organization, label formats, or label contents. Analysis in this chapter enables you to determine if the volume can be processed by your operating system. In some cases, certain modifications may be needed or restrictions observed. If tape volumes are to be transferred permanently to the operating system, use the operating system to create new labels and volume organizations.

IBM Standard Labels

All IBM programming systems create tape labels with the same standard label formats. However, the actual contents of each label field may vary from system to system. To see which fields of each label are functional for the operating system, refer to Figure 8 on page 23, Figure 9 on page 26, Figure 10 on page 33, Figure 16 on page 72, Figure 17 on page 75, and Figure 18 on page 82. Check the processing of these functional fields against the actual contents of the labels you want to use. This comparison should indicate whether the volumes are compatible or what modifications must be made.

Special attention should be given to the data set identifier field of data set label 1 (HDR1, EOVS1, EOF1). The data set name in the label created by another system may contain embedded blanks or special characters. This name is compared to the data set name that you specify in the DD statement; therefore, you must enclose the name in apostrophes on the DD statement that requests this data set. *OS/390 MVS JCL User's Guide* lists the restrictions that apply to enclosing a data set name in apostrophes. The apostrophes do not appear in the data set identifier field.

To match the name in the label, you may have to modify the JFCB after the DD statement information is used to update the JFCB.

The operating system can obtain certain data set characteristics from the standard data set label 2 (HDR2/EOVS2/EOF2). Some IBM programming systems do not use or create data set label 2. The absence of data set label 2 does not interfere with normal processing by the operating system when the label information is specified by some other means. The functional information in data set label 2 (record format, block length, record length, tape recording technique, and printer control characters) can be furnished to the operating system either in the DCB macro or the DD statement.

Labels created by systems other than System/360*, MVS/370, MVS/XA*, or MVS/ESA programming systems should be treated as nonstandard labels, provided that the first record on the tape is not identified as VOL1 and the data sets are followed by recognizable tapemarks.

Nonstandard Labels

Nonstandard labels are labels that do not conform to the formats described in this manual. If you want to retrieve the data set and process the nonstandard labels, you must write nonstandard label processing routines and insert them into the operating system. The procedure is described under Chapter 4, “Nonstandard Labels” on page 119.

If you want to ignore the nonstandard labels, you can retrieve the data set by treating the volume as an unlabeled tape. Use the data set sequence number in the DD statement to bypass the labels and position the tape to the data set.

Unlabeled Tapes

The operating system can process unlabeled tape volumes created by other systems if the data sets are followed by recognizable tapemarks.

Leading Tapemarks

To position a tape at the desired data set, you must specify the correct data set sequence number in the DD statement. If a tapemark precedes the first data set and the LABEL subparameter LTM is specified, then the system tests for and bypasses, if present, a leading tapemark. If a tapemark should precede the first data set and you do not specify LTM in the LABEL parameter field, you must add 1 to the data set sequence number.

If a multivolume data set from another system has a leading tapemark on one or more of the volumes, the operating system can process it as an unlabeled multivolume data set if the LABEL subparameter LTM is specified. Otherwise, the operating system cannot process it as an unlabeled multivolume data set.

The presence of a leading tapemark, that is, a tapemark that precedes the first data set on the tape, makes each data set the second in sequence on the tape. However, the operating system always assumes that continued data sets are first in sequence on the tape. By specifying LTM in the LABEL parameter field, the first data set on a tape can be accessed whether it is preceded by a leading tapemark.

The specification of LTM in the LABEL parameter field does not make allowances for any other excess tapemarks. You must make those adjustments in the data set sequence number.

Appendix A. Using the IBM 3495 Automated Tape Library Dataserver to Read Tape Volumes

Mountable tape cartridges are supported under System Managed Storage with the Tape Library Dataserver. DFSMS provides management facilities for the tape volumes. DFSMSrmm manages data sets on these volumes.

Tape Library

A tape library is a named collection of tape volumes and tape drives. The mounting and demounting of tape cartridges in an automated tape library is performed by robotics in automatic mode and by operators in manual mode. The *Tape Configuration Data Base (TCDB)*, an ICF catalog, contains an inventory of the tape libraries and volumes in the libraries.

Volume Labels

All tape cartridges are labeled with either an IBM or ISO/ANSI standard label. The internal and external labels on a tape cartridge must be unique and equal to one another, and the external label must be machine readable by the vision system in the library. The internal labels on volumes in libraries are automatically labelled by the operating system during the first use. The only valid characters allowed in a volume label are uppercase A-Z and 0-9.

See “System-Managed Mountable Tape External Labels” on page 21, “System-Managed Mountable Tape External Labels” on page 70, and “Cartridge Label” on page 140 for more information on external labels.

Tape Processing

The system does not issue mount (IEC501A and IEC501E) and demount (IEC502E) messages for drives in a Tape Library Dataserver. These messages are recorded to the system log.

Tape Label Verification-Automated Tape Library

Specific Mount Request

The external label is used to verify that the correct volume is mounted. The internal, external, and requested (JCL) volume labels must all match, otherwise the request fails. If there is a label mismatch while the automated tape library is in manual mode, then the volume is demounted and a mount request is reissued. All tape library error retry actions are repeated until the action is successful or a system retry default is exceeded.

Non-Specific Mount Request

A volume's internal and external labels must be identical when the mount is for a non-specific volume. If the labels are not equal or unavailable (no label, uninitialized, etc.), then different recovery actions are taken depending on the mode of operation. If the Tape Library Dataserver is in automatic mode, then the system will:

- Create an internal label if the tape is uninitialized or unlabeled.
- Reject the tape if the internal label is unreadable and RACF Tape Volume Protection is active.

If the Tape Library Dataserver is in automatic mode, but the external label is unreadable, then the system will:

- Create an internal label if the volume is uninitialized.
- Reject the volume if the internal and external labels are unequal, unlabeled, or unreadable, and RACF Tape Volume Protection is active.

If the Tape Library Dataserver is in manual mode, then it will:

- Create an internal label if the volume is uninitialized or unlabeled.
- Issue WTOR IEC704A if the internal and external labels are unequal. The operator can supply the external label or reject the volume.
- Reject the volume if the internal label is unreadable and RACF Tape Volume Protection is active.

If the volume label is extracted from the VOLID MARK and an installation option allows its use in either automatic or manual mode, then the external label is used to verify the volume. The system uses the external label to verify the volume if all of the following are true:

- A 36-track cartridge is mounted in an 18-track drive in response to an output request.
- OPEN or EOVS is able to read the VOLID mark,
- The OPEN or EOVS volume label editor installation exit routine allows this relabelling.
- The user has RACF authority to the volume.
- The DEVSUP member of SYS1.PARMLIB specifies VOLNSNS=YES.

The DEVSUPxx member is described in *OS/390 MVS Initialization and Tuning Reference*.

Tape Label Creation

The issuing of WTOR IEC704A to permit operators to supply label information is the same as for drives outside a Tape Library Dataserver. However, the external volume serial number is displayed in the WTOR. The operator can specify only the 'ownerinfo' and the ANSI access code, not the volume label. This is to keep the internal and external volume serial numbers equal.

See "System-Managed Mountable Tape External Labels" on page 21, "System-Managed Mountable Tape External Labels" on page 70, for more information on external labels.

Tape Library Scratch Tape Pools

Scratch tapes in an automated tape library are kept in a common pool based on cartridge type. The system may select any volume from this group to satisfy a non-specific tape mount. If the tape is rejected during volume verification, then the volume is placed in an error category. These volumes become ineligible for subsequent scratch mount requests until the error is corrected.

Appendix B. Using the IBM 3495 M10 Manual Tape Library Dataserver to Read Tape Volumes

Tape Volumes

Mountable tape cartridges are supported under System Managed Storage with a manual tape library. DFSMS provides management facilities for the tape volumes and data sets on them.

Tape Library

A tape library is a named collection of tape volumes and tape drives. The mounting and demounting of tape cartridges in a manual tape library is performed by operators. The Tape Configuration Data Base (TCDB), an ICF catalog, contains an inventory of the tape libraries and volume in the libraries.

Volume Labels

All tape cartridges are labeled with either an IBM or ISO/ANSI/FIPS standard label. The internal and external labels on a tape cartridge must be unique and equal to one another, and the external label must be machine readable by the vision system in the library. The internal labels on volumes in libraries are automatically labelled by DFP during the first use. The only valid characters allowed in a volume label are uppercase A-Z and 0-9.

See “System-Managed Mountable Tape External Labels” on page 21, “System-Managed Mountable Tape External Labels” on page 70, and “Cartridge Label” on page 140 for more information on external labels.

Message Processing

Messages associated with SMS-managed tape cartridges can be routed to a console name associated with a library and/or one defined with the proper routing codes. Installations can direct messages for a manual tape library by assigning a unique name console to the library. For more information on name console message routing see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Tape Libraries**.

Tape Processing

The system does not issue mount (IEC501A and IEC501E) messages for drives in a manual tape library. The manual tape library manager issues the messages. The mount messages are recorded to the system log and possibly to JOBLOG. The mount order for a specific or scratch tape will be displayed on the library manager's console. Standard and enhanced capacity cartridges are supported in a manual tape library. Installations may assign unique names to identify cartridge types (i.e. MEDIA1) for standard tape cartridges. Then when a mount for a standard scratch cartridge is issued, a mount order for a MEDIA1 will appear on the library manager's console. The mount order will be followed by a list of scratch volumes which

may be used to satisfy the mount request. For additional information on the library manager's console messages, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support*. The demount messages (IEC502E) will be issued to operator consoles with the proper routing codes or named consoles. They will also be recorded to the system log and possibly JOBLLOG.

Tape Label Verification-Manual Tape Library

Specific Mount Request. The internal and requested volume serial numbers will be used to verify that the correct volume is mounted. If there is a mismatch between the volume serial numbers, the volume will be demounted and a mount request reissued. All library error retry actions are repeated until the action is successful or a system retry default is exceeded.

Non-Specific Mount request. The internal volume serial number will be used to verify its usage in a manual tape library. The volume serial number must be defined to the specific manual tape library and its use attribute must be defined as a scratch. If the scratch volume fails verification, it will be demounted and a mount request reissued for another scratch volume. If a volume's internal label is unavailable (no label, not initialized, etc.), the operator will be prompted for the volume's external volume serial number. The volume serial number supplied by the operator will be verified prior to labeling the tape volume. The supplied volume serial number must be known to the manual library as a scratch volume. If the volume label is extracted from the VOLID MARK, and an installation option allows its use, it will be used during volume verification.

Tape Library Scratch Tape Pools

Any scratch tape in a manual tape library may be used to satisfy a non-specific mount request. If a specific cartridge type is requested, then any scratch tape of that cartridge type may be used. Any tape rejected during volume verification will be placed in an error category. These volumes become ineligible for subsequent scratch mount request until the error is corrected.

Appendix C. Component Support of Label Processing

Job control statements make the label processing facilities of data management available to users of the operating system's assembler, linkage editor, Sort/Merge program product, utility programs, and high-level language program products.

Figure 23 shows the component support for each type of label processing.

Figure 23. Component Support of Label Processing Types

Type	Assembler	Linkage Editor	DFSORT*	Utilities	COBOL*			IBM Compilers
					AD/CYCLE* COBOL/370*	VS COBOL/2*	OS/VS* COBOL 2.4	
Uses Data Management Facilities for Label processing	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Supports Standard Labels (SL, AL)	Yes	Yes	Yes	SL-Yes AL-IEHINITT AL-IEBGENER	Yes	Yes	Yes	Yes
Supports Standard User Labels (SUL, AUL)	No	No	Yes	Yes	SUL-Yes AUL-No	SUL-Yes AUL-No	SUL-Yes AUL-Yes	No
Supports Nonstandard Labels (NSL) 1	Yes	Yes	Yes	Yes	No	No	Yes	Yes
Supports Unlabeled Tapes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Supports Bypass Label Processing Option 2	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Supports Concatenated Data Sets with Unlike Attributes	No	Yes	Yes	No	No	No	No	No

Notes:

1. NSL can be specified only when installation—written routines that write and process the nonstandard labels have been incorporated into the operating system.
2. If the BLP option is not allowed by the system, its use defaults to NL.

Appendix D. External Labels

External labels are affixed to the tape reels and cartridges to provide visual identification of the volume and its contents. Normal tape volume control may require several types of external tape labels. One is a *permanent label* that identifies the reel; the other is a *temporary label* that identifies the contents. A third type of label, the *checkpoint security label*, may be applied to tape reels or cartridges that contain checkpoint data sets.

To write on external labels, you should use a pen or a felt-tip marker that does not produce loose residue. Do not use a lead pencil. Do not use an eraser.

Reel Label

The reel label should be applied with a permanent-type adhesive, so that it cannot be easily removed. It is affixed when the tape is first received by your installation. The label should contain the sequential volume serial number assigned by your installation; it may also identify your installation. The volume serial numbers are used to identify the tape reel by a unique number and to file the tapes in the tape rack. An example of a reel label is shown in Figure 24.



Figure 24. External Label for Reel Identification

Contents Label

The contents label is used to identify the current contents of a particular volume. Because this is a temporary label, it should be applied with adhesive that is strong enough to hold the label securely and yet allow easy removal.

This label is applied when data is written on the volume and contains identifying information to ensure that the contents of the volume can be easily distinguishable from others. Your installation determines the format of the label. The information entered in the label is usually furnished partly by the programmer and partly by the operator. Examples of contents labels are shown in Figure 25 on page 140.

REEL NUMBER			PROGRAMMER'S NAME, DEPT., BLDG		
				DATE	
			SCRATCH DATE	SYSTEM	
DENSITY	PARITY	TRACK	TAPE DESCRIPTION		

DESC.		JOB#		
REEL NO.	CREATN DATE	RET CYC	OP NO	FILE IDENTIFIER

Figure 25. External Labels for Contents Identification

Checkpoint Security Label

A checkpoint security label may be put on tape volumes containing checkpoint data sets. This label, applied by the operator, helps ensure security of the checkpoint data set.

The checkpoint security label is a temporary label; it should be applied with adhesive strong enough to hold the label securely and yet allow easy removal of the label later when the checkpoint data set is deleted. The size and placement of the label should not interfere with the handling of the tape.

Cartridge Label

An automated tape library cartridge label can have a maximum of six uppercase characters for identification purposes:

Uppercase A-Z

Numerics 0-9

Trailing blank space (non-service volumes only)

Note: The first three characters can be either alpha or numeric, but the last three characters must be numeric. Blanks, if used, must follow the last character.

The external label must be applied within the indentation on the side of the cartridge; however, a larger information label, supplied by the customer, can be used on the top of the cartridge. An example of a cartridge label is shown in Figure 26 on page 141. See "System-Managed Mountable Tape External Labels" on page 21, "System-Managed Mountable Tape External Labels" on page 70, "Volume Labels" on page 131, and "Tape Label Verification-Automated Tape Library" on page 131 for more information on the cartridge external label.

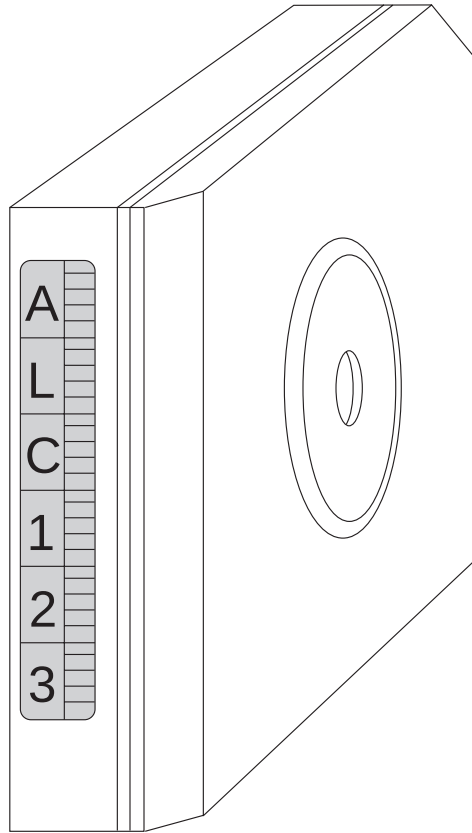


Figure 26. External Label for Cartridge Identification

Appendix E. Equivalent ASCII and EBCDIC Codes

ASCII

The routine included in the system converts ASCII 7-bit code to and from EBCDIC.

All EBCDIC codes that convert to an ASCII code that cannot be contained in seven bits are represented by the substitute character X'1A'.

The following tables show the relationship of EBCDIC code to ASCII 8-bit code.

Note: This conversion is not necessarily reversible. The ASCII characters that cannot be converted are changed to EBCDIC substitute characters (X'3F'). The EBCDIC characters that cannot be converted are changed to ASCII substitute characters (X'1A').

EBCDIC to ASCII

EBCDIC (hex)	ASCII (hex)	EBCDIC Meaning	EBCDIC (hex)	ASCII (hex)	EBCDIC Meaning
00	00	NUL	24	1A	BYP/INP
01	01	SOH	25	0A	LF
02	02	STX	26	17	ETB
03	03	ETX	27	1B	ESC
04	1A	SEL	28	1A	SA
05	09	HT	29	1A	SFE
06	1A	RNL	2A	1A	SM/SW
07	7F	DEL	2B	1A	CSP
08	1A	GE	2C	1A	MFA
09	1A	SPS	2D	05	ENQ
0A	1A	RPT	2E	06	ACK
0B	0B	VT	2F	07	BEL
0C	0C	FF	30	1A	
0D	0D	CR	31	1A	
0E	0E	SO	32	16	SYN
0F	0F	SI	33	1A	IR
10	10	DLE	34	1A	PP
11	11	DC1	35	1A	TRN
12	12	DC2	36	1A	NBS
13	13	DC3	37	04	EOT
14	1A	RES/ENP	38	1A	SBS
15	1A	NL	39	1A	IT
16	08	BS	3A	1A	RFF
17	1A	POC	3B	1A	CU3
18	18	CAN	3C	14	DC4
19	19	EM	3D	15	NAK
1A	1A	UBS	3E	1A	
1B	1A	CU1	3F	1A	SUB
1C	1C	IFS	40	20	(space)
1D	1D	IGS	41	1A	RSP
1E	1E	IRS	42	1A	
1F	1F	ITB/IUS	43	1A	
20	1A	DS	44	1A	
21	1A	SOS	45	1A	
22	1A	FS	46	1A	
23	1A	WUS	47	1A	
			48	1A	
			49	1A	

EBCDIC (hex)	ASCII (hex)	EBCDIC Meaning	EBCDIC (hex)	ASCII (hex)	EBCDIC Meaning
4A	5B	¢	83	63	c
4B	2E	.	84	64	d
4C	3C	<	85	65	e
4D	28	(	86	66	f
4E	2B	+	87	67	g
4F	21		88	68	h
50	26	&	89	69	i
51	1A		8A	1A	
52	1A		8B	1A	
53	1A		8C	1A	
54	1A		8D	1A	
55	1A		8E	1A	
56	1A		8F	1A	
57	1A		90	1A	
58	1A		91	6A	j
59	1A		92	6B	k
5A	5D	!	93	6C	l
5B	24	\$	94	6D	m
5C	2A	*	95	6E	n
5D	29	)	96	6F	o
5E	3B	;	97	70	p
5F	5E	¬	98	71	q
60	2D	—	99	72	r
61	2F	/	9A	1A	
62	1A		9B	1A	
63	1A		9C	1A	
64	1A		9D	1A	
65	1A		9E	1A	
66	1A		9F	1A	
67	1A		A0	1A	
68	1A		A1	7E	°
69	1A		A2	73	s
6A	7C		A3	74	t
6B	2C	,	A4	75	u
6C	25	%	A5	76	v
6D	5F	—	A6	77	w
6E	3E	>	A7	78	x
6F	3F	?	A8	79	y
70	1A		A9	7A	z
71	1A		AA	1A	
72	1A		AB	1A	
73	1A		AC	1A	
74	1A		AD	1A	
75	1A		AE	1A	
76	1A		AF	1A	
77	1A		B0	1A	
78	1A		B1	1A	
79	60		B2	1A	
7A	3A	:	B3	1A	
7B	23	#	B4	1A	
7C	40	@	B5	1A	
7D	27	'	B6	1A	
7E	3D	=	B7	1A	
7F	22	"	B8	1A	
80	1A		B9	1A	
81	61	a	BA	1A	
82	62	b	BB	1A	

EBCDIC (hex)	ASCII (hex)	EBCDIC Meaning
BC	1A	
BD	1A	
BE	1A	
BF	1A	
C0	7B	
C1	41	A
C2	42	B
C3	43	C
C4	44	D
C5	45	E
C6	46	F
C7	47	G
C8	48	H
C9	49	I
CA	1A	
CB	1A	
CC	1A	
CD	1A	
CE	1A	
CF	1A	
D0	7D	
D1	4A	J
D2	4B	K
D3	4C	L
D4	4D	M
D5	4E	N
D6	4F	O
D7	50	P
D8	51	Q
D9	52	R
DA	1A	
DB	1A	
DC	1A	
DD	1A	
DE	1A	
DF	1A	
E0	5C	\
E1	1A	
E2	53	S
E3	54	T
E4	55	U
E5	56	V
E6	57	W
E7	58	X
E8	59	Y
E9	5A	Z
EA	1A	
EB	1A	
EC	1A	
ED	1A	
EE	1A	
EF	1A	
F0	30	0
F1	31	1
F2	32	2
F3	33	3
F4	34	4

EBCDIC (hex)	ASCII (hex)	EBCDIC Meaning
F5	35	5
F6	36	6
F7	37	7
F8	38	8
F9	39	9
FA	1A	
FB	1A	
FC	1A	
FD	1A	
FE	1A	
FF	1A	

ASCII to EBCDIC

ASCII (hex)	EBCDIC (hex)	ASCII Meaning
00	00	NUL
01	01	SOH
02	02	STX
03	03	ETX
04	1A	SEL
05	09	HT
06	1A	RNL
07	7F	DEL
08	1A	GE
09	1A	SPS
0A	1A	RPT
0B	0B	VT
0C	0C	FF
0D	0D	CR
0E	0E	SO
0F	0F	SI
10	10	DLE
11	11	DC1
12	12	DC2
13	13	DC3
14	3C	DC4
15	3D	NAK
16	32	SYN
17	26	ETB
18	18	CAN
19	19	EM
1A	3F	SUB
1B	27	ESC
1C	1C	FS
1D	1D	GS
1E	1E	RS
1F	1F	US
20	40	(space)
21	4F	!
22	7F	"
23	7B	#
24	5B	\$
25	6C	%
26	50	&
27	7D	'
28	4D	(
29	5D	)
2A	5C	*
2B	4E	+
2C	6B	,
2D	60	-
2E	4B	.
2F	61	/
30	F0	0
31	F1	1
32	F2	2
33	F3	3
34	F4	4
35	F5	5
36	F6	6

ASCII (hex)	EBCDIC (hex)	ASCII Meaning
37	F7	7
38	F8	8
39	F9	9
3A	7A	:
3B	5E	;
3C	4C	<
3D	7E	=
3E	6E	>
3F	6F	?
40	7C	@
41	C1	A
42	C2	B
43	C3	C
44	C4	D
45	C5	E
46	C6	F
47	C7	G
48	C8	H
49	C9	I
4A	D1	J
4B	D2	K
4C	D3	L
4D	D4	M
4E	D5	N
4F	D6	O
50	D7	P
51	D8	Q
52	D9	R
53	E2	S
54	E3	T
55	E4	U
56	E5	V
57	E6	W
58	E7	X
59	E8	Y
5A	E9	Z
5B	4A	[
5C	E0	\
5D	5A	]
5E	5F	^
5F	6D	_
60	79	`
61	81	a
62	82	b
63	83	c
64	84	d
65	85	e
66	86	f
67	87	g
68	88	h
69	89	i
6A	91	j
6B	92	k
6C	93	l
6D	94	m
6E	95	n
6F	96	o

ASCII (hex)	EBCDIC (hex)	ASCII Meaning
70	97	p
71	98	q
72	99	r
73	A2	s
74	A3	t
75	A4	u
76	A5	v
77	A6	w
78	A7	x
79	A8	y
7A	A9	z
7B	C0	{
7C	6A	
7D	D0	}
7E	A1	~
7F	07	DEL
80-FF	3F	

Conversion Irregularities

The following irregularities exist in the graphic character set between EBCDIC and ASCII 7-bit and 8-bit codes:

EBCDIC Code		8-Bit Code		7-Bit Code	
Graphic	Hex	Graphic	Hex	Graphic	Hex
¢	X'4A'	[	X'5B'	[	X'5B'
!	X'5A'	]	X'5D'	]	X'5D'
[	X'AD'	(n/a)	X'D5'	Sub	X'1A'
]	X'BD'	(n/a)	X'E5'	Sub	X'1A'
	X'4F'	!	X'21'	!	X'21'
¡	X'6A'	¡	X'7C'	¡	X'7C'

For example, an exclamation mark is coded in EBCDIC as X'5A', but in ASCII 7-bit and 8-bit codes as X'21'.

Abbreviations

The following abbreviations are defined as they are used in the DFSMS/MVS library. If you do not find the abbreviation you are looking for, see *Dictionary of Computing*, New York: McGraw-Hill, 1994.

This list may include acronyms and abbreviations from:

- The *American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018.
- The *Information Technology Vocabulary* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1).

A

ACS. Automatic class selection.

AL. American National Standard Labels.

ASCII. American National Standard Code for Information Interchange.

AUL. ANSI and user header or trailer labels.

B

BCDIC. Binary coded decimal interchange code.

BDW. Block descriptor word.

BLP. Bypass label processing.

BPI. Bytes per inch.

C

CVOL. Control volume.

CVT. Communication vector table.

D

DD. Data definition.

DDR. Dynamic device reconfiguration.

DECB. Data event control block.

DOM. Delete operator message.

DSCB. Data set control block.

DFSORT. Data Facility Sort.

E

EBCDIC. Extended binary coded decimal interchange code.

EOF1. Standard data set label 1 trailer label.

EOF2. Standard data set label 2 trailer label.

EOV1. Standard data set label 1 trailer label.

EOV2. Standard data set label 2 trailer label.

F

FEOV. force end of volume

FIPS. Federal Information Processing Standard.

G

GCR. Group coded recording.

GDG. Generation data group.

GDS. Generation data set

H

HDR1. Standard data set label 1 header label.

HDR2. Standard data set label 2 header label.

I

IDRC. Improved Data Recording Capability

IOB. Input/output block.

IRG. Interrecord gap.

J

JFCB. Job file control block.

L

LTM. Leading tapemark.

M

MPF. Message Processing Facility.

N

NL. No label.

NSL. Nonstandard label.

O

O/C/EOV. Open/close/end-of-volume.

R

RDBACK. Read backward.

S

SKP. Skip sequential processing.

SL. IBM standard label.

SML. Storage Management Library.

SMS. Storage Management Subsystem.

SSL. Storage Subsystem Library.

SUL. IBM standard and user header or trailer label.

T

TCB. Task control block.

| **TCDB.** Tape configuration data base.

TIOT. Task I/O table.

TM. Tapemark.

TTR. Track record address.

U

UCB. Unit control block.

UHL. User header label.

UPD. Update mode.

UTL. User trailer label.

UVL. User volume label.

V

VOL1. Standard volume label.

VOLSER. Volume serial number.

Glossary

The following terms are defined as they are used in the DFSMS/MVS library. If you do not find the term you are looking for, see *Dictionary of Computing*, New York: McGraw-Hill, 1994.

This glossary may include terms and definitions from:

- The *American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018.
- The *Information Technology Vocabulary* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1).

A

allocation. Generically, the entire process of obtaining a volume and unit of external storage, and setting aside space on that storage for a data set.

automatic class selection (ACS). A mechanism for assigning SMS classes and storage groups to data sets.

B

backup data set. A copy that can be used to replace or reconstruct a damaged data set.

D

data class. A list of the data set allocation parameters and their values, used when allocating a new SMS-managed data set in an MVS operating environment.

E

entry. A collection of information about a cataloged object in a master or user catalog. Each entry resides in one or more 512-byte records.

entry name. A unique name for each component or object as it is identified in a catalog. The entry name is the same as the dsname in a DD statement that describes the object.

G

generation data group base entry. An entry that permits a non-VSAM data set to be associated with other non-VSAM data sets as generation data sets.

gigabyte. 1073741824 bytes.

header label. Data set labels that precede a data set on tape.

L

leading tapemark bypass (LTM). Code signaling that if a leading tapemark is encountered on an unlabeled tape, it is bypassed.

M

management class. A list of the migration, backup, and retention parameters and their values, for an SMS-managed data set.

MVS/ESA. An MVS operating system environment which supports ESA/370.

O

OS control volume (OS CVOL). A type of catalog commonly used on very old levels of MVS.

R

reflective strip. A physical signal that marks the logical end of 7-track and 9-track reels of tape. Newer IBM magnetic tape subsystems, like the IBM 3480, use tape cartridges instead of tape reels. These subsystems have internal mechanisms that sense the beginning and end of tape, and do not use reflective strips.

S

simple name. The rightmost component of a qualified name. For example, APPLE is the simple name in TREE.FRUIT.APPLE. The simple name corresponds to the lowest index level in the catalog for the data set name.

SMS class. A list of attributes that SMS applies to data sets having similar allocation (data class), performance (storage class), or backup and retention (management class) needs.

SMS-managed data set. A data set that has been assigned a storage class.

storage administrator. A person in the data processing installation who is responsible for defining, implementing, and maintaining storage management policies.

storage class. A list of DASD storage performance, security, and availability service level requirements for an SMS-managed data set.

Storage Management Subsystem (SMS). An operating environment that helps automate and centralize the management of storage. To manage storage, SMS provides the storage administrator with control over data class, storage class, management class, storage group, and ACS routine definitions.

T

task control block (TCB). Holds control information related to a task.

translator. A tape device feature that takes 8-bit EBCDIC characters from the buffer and writes them as 6-bit BCD characters. It translates the other way for read operations. For 7-track tapes.

U

unit control block (UCB). A control block in storage that describes the characteristics of a particular I/O device on the operating system.

user labels. Data set label groups that can include standard user labels on tape.

Index

Numerics

- 3430 Magnetic Tape Subsystem
 - tape characteristics 13
- 3480 Magnetic Tape Subsystem 15
 - See also* tape, cartridge
- 7-track feature
 - code translation 13, 15
 - data conversion feature 13
 - density requirements 14
 - lack of ISO/ANSI support 14
 - models allowing 13
- 9-track tape 13

A

- accessibility field
 - ISO/ANSI data set 1 label 97
 - ISO/ANSI standard data set label 1 79
 - ISO/ANSI standard volume label 73
 - ISO/ANSI VOL1 label 96
- AL subparameter 2
- allocating
 - generation data set 6
 - tape data set 3
- ASCII 143
 - See also* ASCII interchange tape
 - control character 37
 - conversion codes 143, 147
- ASCII interchange tape
 - conversion from EBCDIC 99, 143
- assembler support 137
- assigning
 - volume serial number
 - DFSMSrmm tool 73
 - IEHINITT utility program 24, 25, 73
 - JCL statement 24, 25, 73
 - system assignment 126
- AUL subparameter 2
- automated tape library 21, 24, 40, 48, 58, 90, 98, 131
- AVR (automatic volume recognition) 10, 121

B

- backward read xv
 - See also* RDBACK parameter
- basic tape format 1, 2
- BCDIC 13, 17, 64
- binary data 14
- bits per inch 13
 - See also* density, bpi
- blank tape
 - bypass label processing 11

- blank tape (*continued*)
 - IEHINITT utility
 - IBM standard label 42
 - ISO/ANSI standard label 92
- block
 - count
 - IBM standard label 31, 48, 50
 - ISO/ANSI standard data set label 1 80
 - ISO/ANSI standard label 107, 109
 - verification 109
 - length
 - description 35
 - ISO/ANSI standard data set label 2 84
 - padding 101
- BLP (bypass label processing)
 - component support 137
 - processing 11, 121
 - specification 2
- BLP subparameter 2, 11
- bpi (bits per inch) 13
 - See also* density, bpi
- buffer
 - alignment block 86
 - offset 86
- bypass label processing 2
 - See also* BLP

C

- cartridge 15
 - See also* tape, cartridge
- cataloged data set
 - description 5
 - generation group 5
 - IBM standard label 45, 51
 - ISO/ANSI standard label 104, 109
 - no label 123
 - retrieve 5
- CCSID 86
- character code, EBCDIC 14
- checkpoint/restart xv, 9, 38, 61, 101, 140
 - See also* restart routine
- CLOSE macro
 - function 9
 - IBM standard label 53, 60
 - ISO/ANSI standard label 117
 - ISO/ANSI/ standard label 111
 - no label 127
- close routine
 - function 9
 - IBM standard label 53, 60
 - ISO/ANSI standard label 117

- close routine (*continued*)
 - ISO/ANSI/ standard label 111
 - no label 127
- COBOL 137
- compacting data 15
- component support 137
- concatenation
 - data set
 - description 7
 - IBM standard label 49, 51, 53, 110
 - ISO/ANSI standard label 97, 108, 111
 - no label 124
 - size order 7
- connected data set xv
 - See also* concatenation, data set
- contents label 139
- control character
 - IBM standard label 37, 49
 - ISO/ANSI standard label 85
- conversion
 - ASCII to/from EBCDIC 99, 143
 - codes 143
 - ASCII 147
 - EBCDIC 143
 - data
 - BCDIC to/from EBCDIC 14
- creation date
 - IBM standard label 29
 - ISO/ANSI standard data set label 1 78

D

- data class 3
- data control block xv
 - See also* DCB
- data conversion 14, 99, 143, 147
- data group index 5
- data management routine 9
- data recording characteristics 12, 15
- data set
 - accessibility 97
 - allocate new generation 6
 - allocating 3
 - attributes 3, 8
 - cataloged 5
 - description 5
 - retrieve data from 5
 - characteristics 48
 - concatenated 53, 111
 - See also* concatenation, data set
 - DASD advantages 5
 - description 3
 - expiration date 3
 - forward merge 4
 - generation 5
 - See also* GDS

- data set (*continued*)
 - header label
 - IBM ISO/ANSI 70
 - IBM standard label 20, 22
 - ISO/ANSI standard label 68
 - identifier 27, 129
 - information
 - device and structure recorded on 5
 - recorded 5
 - managing tapes using DFSMSrmm 5
 - multiple on multiple volumes 8
 - multiple on single volume 7
 - multiple per volume 7
 - multiple volumes per data set 7, 50, 53, 109, 111, 116
 - opening 45
 - pass job step 7
 - passed 7
 - position
 - IBM standard label 2, 36
 - ISO/ANSI standard label 2
 - ISO/ANSI standard labeled tape 85
 - processing method 8
 - proper retrieval 7
 - reverse merge 4
 - security protection 3
 - serial number
 - IBM standard label 28
 - SMS-managed 5
 - specifying sequence number 8
 - volume label 1
 - writing header labels 58
- data set label 1 (HDR1, EOVI, EOF1)
 - IBM standard
 - contents 25, 31
 - define 17, 19
 - format 25, 31
 - other systems 129
 - processing 25, 31, 40, 61
 - ISO/ANSI standard
 - contents 74, 80
 - define 63, 70
 - format 74, 80
 - processing 74, 80, 91
 - writing 114
- data set label 2 (HDR2, EOVI, EOF2)
 - IBM standard
 - contents 32, 38
 - define 17, 19
 - format 32, 38
 - other systems 129
 - processing 32, 38, 40, 61
 - ISO/ANSI standard
 - writing 114
 - ISO/ANSI standard
 - contents 81, 86
 - define 63, 70

- data set label 2 (HDR2, EOVS, EOF2) (*continued*)
 - ISO/ANSI standard (*continued*)
 - format 81, 86
 - processing 81, 86, 91, 107
 - writing 116
- data set name
 - description 47, 106
 - IBM standard label
 - checking before input processing 49
 - checking before output processing 57, 58
 - format in HDR1 label 27
 - ISO/ANSI standard label
 - checking before output processing 98
 - format in HDR1 label 76
 - ISO/ANSI/standard label
 - duplicate name checking 106
 - other systems 129
- data set protection
 - description 100
 - ISO/ANSI standard label 94
 - multiple volumes
 - IBM standard label 52, 53
 - ISO/ANSI standard label 111
 - password protection 48, 98
 - processing
 - IBM standard label 30, 31, 45, 60
 - ISO/ANSI standard label 93
 - specification 2, 3
- data set sequence number
 - concatenated data set 53, 111
 - See also* concatenation, data set
 - default 3
 - IBM standard label 28, 45, 55
 - ISO/ANSI standard label 104, 113, 116
 - description 77
 - multiple volumes
 - IBM standard label 53
 - ISO/ANSI standard label 111
 - no label 123, 126
 - other systems 130
 - repositioning during Restart 61
 - specification 3
 - SYSOUT data set 55
- data set sequence subparameter 3
- DATACLAS attribute 6
- DATACLAS keyword 3
- DCB (data control block) 59
 - completion 3, 5
 - DATACLAS attribute 6
 - end-of-data routine
 - IBM standard label 49, 53
 - ISO/ANSI standard label 108, 111
 - no label 124
 - exit list 20, 21, 39, 49
 - defer label processing 53
 - UHL 58

- DCB (data control block) (*continued*)
 - exit routine 4
 - IBM user label exit 49, 50, 61
 - ISO/ANSI user label exit 108, 115
 - LIKE attribute 6
 - macro
 - completing data control block 3, 5
 - DEN parameter 13, 15
 - TRTCH parameter 14
 - merging information 4, 5
 - REFDD attribute 6
- DCBOFPPC bit 51, 110
- DD statement
 - concatenated data set 7
 - DCB parameter 6
 - DATACLAS attribute 6
 - LIKE attribute 6
 - REFDD attribute 6
 - DEN parameter 13, 15
 - DISP parameter 10
 - LABEL parameter 2
 - multiple data sets 7, 8
 - multiple volumes 7
 - parameter functions 3
 - passed data set 7
 - TRTCH parameter 14
 - VOLUME parameter 7
- DDR (dynamic device reconfiguration)
 - volume verification 119
- deferred user trailer label processing 59, 61, 108
 - See also* trailer label
 - IBM standard label 41
- DEN parameter
 - description 13, 15
- density
 - bpi 13, 15
 - conflict 13
 - DEN parameter codes 14
 - IBM standard label 35
 - ISO/ANSI standard label 85
 - tape cartridge 15
- DFSMSrmm (DFSMS removable media manager)
 - managing tapes 5
- DFSORT (Data Facility Sort) 137
- DISP parameter
 - description 10
- DISP=MOD 4
 - checking labels for output tape 40
- disposition
 - tape positioning parameters 10
- dual density 13
- dummy header label
 - IBM standard label 41
 - ISO/ANSI standard label 92, 114
- dynamic device reconfiguration 119
 - See also* DDR

E

EBCDIC (extended binary coded decimal interchange code)
 conversion codes 143, 147
editor routines 43, 94, 119
EMODVOL1 module 120
end of data set
 description 9
 IBM standard label 49, 53
 ISO/ANSI standard label 108, 111
 no label 124
end of reel 15
end-of-data 9
 See also EODAD routine
end-of-volume routine xv
 See also EOVR routine
EODAD (end-of-data) routine
 description 9
 IBM standard label 49, 53
 ISO/ANSI standard label 108, 111
 no label 124
EOVR (end-of-volume) routine
 data set label 1 xv
 See also data set label 1
 data set label 2 xv
 See also data set label 2
 description 9
 IBM standard label 49, 52, 60, 117
 ISO/ANSI standard label 58, 108, 111, 115
 macro 9
 no label 124
 OPTCD=B 27
 special conditions 60, 117
 trailer label 76
 volume label editor routine 43, 94, 119
error
 permanent I/O
 ISO/ANSI standard label 115
 processing 58
exit list 50
exit routine 47, 52, 54, 56, 59
 See also Open/EOVR user exit
expiration date
 existing labels 48, 107
 IBM standard label 29, 48, 107
 ISO/ANSI standard data set label 1 78
 ISO/ANSI standard label 114
 LABEL parameter 3
external label 139

F

FEOVR macro
 IBM standard label
 input tape 41, 49, 50, 53, 109
 output tape 41

FEOVR macro (*continued*)

 ISO/ANSI standard label
 input tape 108
 output tape 58, 108, 115
 no label 124, 127
file identifier 76
file section number 77
file sequence number 77
 See also data set sequence number
file set identifier 77
first record, verification
 bypass label processing 11
 IBM standard label 45, 52
 ISO/ANSI standard label 104, 111
 no label 123, 125
format 25, 31, 38, 39, 74, 86, 87, 89
 See also data set label 1, data set label 2, and user label
FORTRAN 137

G

GDG (generation data group)
 description 5
 IBM standard label 28
 ISO/ANSI standard label 78
GDS (generation data set)
 allocating 6
generation data set 6
 See also GDS
generation number
 description 5
 IBM standard label 28
 ISO/ANSI standard data set label 1 78

H

HDR1 label xv
 See also data set label 1
HDR2 label xv
 See also data set label 2
header label
 data set 20, 68
 defined 1
 IBM identifiers 17
 ISO/ANSI identifiers 64
 user 20
high speed search 55
 See also tape

I

I/O support routines 9
 See also close routine, EOVR routine, open routine
IBM standard label
 component support 137

- IBM standard label (*continued*)
 - defined 17
 - defined specification 1
 - operating system support 17
 - overview of processing 40, 41
 - types 17
 - volume layout 17, 19
- IEC.TAPERING 44, 96
- IEHINITT program
 - assigning volume serial numbers 24, 25
 - creating IBM standard labels 41
 - creating ISO/ANSI standard labels 92, 96
 - dummy IBM HDR1 label 41
 - dummy ISO/ANSI HDR1 label 114
 - dummy ISO/ANSI/ HDR1 label 92
- Improved Data Recording Capability 15
- IN subparameter 3
- input data set
 - closing 111
 - IBM standard label 45, 53
 - ISO/ANSI standard label 103, 111
 - no label 122, 125
 - opening 103
 - specifying 8
- installation exit
 - defaults 99
 - ISO/ANSI standard label 99, 101
 - RACHECK 101
 - WTOR 99
- International Organization for Standardization xv
 - See also* ANSI standard label
- ISO (International Organization for Standardization) xv
 - See also* ANSI standard label
- ISO/ANSI standard label
 - characteristics 99, 102
 - compared to IBM standard label 64
 - component support 137
 - defined 64
 - header label 68
 - See also* data set label 1, data set label 2
 - installation exits 99, 101
 - overview of processing 90
 - specification 1
 - tapemark 103
 - trailer label 69
 - See also* data set label 1, data set label 2
 - types 64
 - Version 1
 - defined 63
 - Version 3
 - defined 63
 - volume label
 - creation 68, 92
 - defined 68
 - format 71, 74
 - processing 91

- ISO/ANSI standard label (*continued*)
 - volume organization 65
- ISO/ANSI/ standard label
 - Version 3
 - processing on another MVS system 102
- ISO/ANSI/standard label
 - protecting data through volume accessibility 94

J

- JFCB (job file control block) 4
 - changing information in the JFCB 4
 - data set sequence number xv
 - See also* positioning tapes
 - header label information 46, 49
 - merging control block information 45, 49
 - merging information into the JFCB 4
 - updating 3, 5
 - volume serial number 22, 25, 51, 54
- job and job step name
 - IBM data set label 2 36
 - ISO/ANSI data set label 2 85
- job control statement 2, 8
- job file control block xv
 - See also* JFCB

K

- keyword
 - DATACLAS 3

L

- label
 - anomaly exit 112
 - checkpoint security 140
 - definition and organization 17, 22, 70
 - editor routine 43, 94, 119
 - format 17, 25
 - label identifier
 - data set label 1 25
 - nonstandard 130
 - processing
 - IBM standard label 40, 41
 - ISO/ANSI standard label 90, 104
 - system components 137
 - standard level 74
 - tape
 - define 1, 2, 3
 - external 21, 139
 - model 2
 - requirements 1
 - types 1, 3
 - type 2
 - validation 100

- label identifier
 - IBM standard label
 - data set label 39
 - data set label 1 26
 - data set label 2 34
 - volume label 23
 - IBM user label 39
 - ISO/ANSI standard label
 - data set label 75, 82
 - user label 87
 - volume label 72
- label number
 - IBM standard label
 - data set label 1 27
 - data set label 2 34
 - user label 39
 - volume label 24
 - ISO/ANSI standard label
 - data set label 76, 83
 - user label 87
 - volume label 73
 - ISO/ANSI/ standard label
 - user label 88
- LABEL parameter 2
 - AL subparameter 2
 - AUL subparameter 2
 - BLP subparameter 2
 - data set sequence subparameter 3
 - description 2
 - expiration date 3
 - IN subparameter 3
 - LTM subparameter 2
 - NL subparameter 2
 - OUT subparameter 3
 - security protection 3
 - SL subparameter 2
 - SUL subparameter 2
- leading tape mark 2, 130
- LEAVE parameter
 - description 10
 - IBM standard label 46
 - ISO/ANSI standard label 105
 - no label 123
- library 21, 24
- LIKE keyword 6
- linkage editor
 - label processing 137
- load point 15
- logical write-protection 44, 95
- lookahead mounting 52, 59, 110, 115
- LPALIB
 - volume verification routines 119
- LTM subparameter 2, 130

M

- magnetic tape
 - characteristics 12, 15
 - label xv
 - See also* label, tape
- member
 - SVC library 119
- merging information into the JFCB 4
- messages
 - tape
 - mount 52
 - ring 45, 96
- model DSCB 2, 6
- module name
 - nonstandard label routines 119
 - volume label
 - editor routine 120
- mount switch 45, 54, 104
 - See also* UCBDMCT mount switch
- mounting volumes 10
- multiple data sets
 - DD statement 7
 - IBM standard label
 - EOV conditions 60, 117
 - input data set 45, 49
 - output data set 55, 58
 - restart procedure 61
 - volume organization 17, 19
 - ISO/ANSI standard label
 - input data set 104, 106
 - output data set 114
 - volume organization 63, 66
 - no label 123, 124
 - nonstandard label 119
- multiple volumes
 - DD statement 7
 - IBM standard label
 - checking volume labels 45, 52, 53
 - creating volume labels 41
 - from another system 129
 - switching volumes 50
 - volume organization 17, 19
 - ISO/ANSI standard label
 - checking volume labels 104, 111
 - creating volume labels 93
 - switching volumes 58, 109, 115
 - volume organization 63, 66
 - no label 124, 125
 - nonstandard label 119
 - RACF protection 8

N

- named generation group 5

- new volume label
 - IBM standard volume label 41
 - ISO/ANSI standard volume label 116
- NL subparameter 2, 121, 123
- nonspecific volume 7
- nonstandard label
 - component support
 - features 137
 - other systems 130
 - specification 1
 - processing routines xv
 - See also* RACF nonstandard label
 - routine, module name 119
- NRZI mode 13
- NSL subparameter 2, 130
- NSLREPOS routine 119

O

- OMODVOL1 module 120
- OPEN macro
 - overriding INOUT, OUTIN, and OUTINX parameters 3
 - processing for a tape data set 9, 12
- open routine
 - function of 9
 - IBM standard label 45, 49, 53, 58
 - ISO/ANSI standard label 103, 108, 112, 115
 - no label 122, 127
- open, volume label editor routine 43, 94, 119
- Open/EOV user exit
 - nonspecific tape mount request 54
 - security verification 47, 52, 56, 59
- opening
 - input data set
 - IBM standard label 45, 49
 - ISO/ANSI standard label 103, 108
 - no label 122, 124
 - output data set
 - caution 53
 - caution—empty data sets 125
 - IBM standard label, specified 53, 58
 - ISO/ANSI standard label, specified 112, 115
 - no label specified 125
- OPTCD=B (disregard EOF) 51, 110
- OUT subparameter 3
- output
 - data set
 - closing 60, 127
 - IBM standard label 53, 61
 - ISO/ANSI standard label 112, 118
 - no label 125, 127
- owner identification
 - IBM standard volume label 25
 - ISO/ANSI standard volume label 74

P

- parallel mounting 52, 59, 110, 115
- parity 12, 15
- passed data set attributes 7
- password
 - checking 48
 - protection xv
 - See also* data set protection
 - description 48, 57
- password protection 45
- phase encoded mode 13
- PL/I 137
- positioning
 - tape
 - IBM standard label 45, 47, 55, 56
 - ISO/ANSI standard label 104, 106, 113
 - no label 123, 125, 126
 - OPEN and CLOSE parameters 10
 - volume to data set 104
- processing
 - component considerations 137
 - HDR2 label 107
 - routines 8
 - user header label 108
 - version differences 102
- program properties 43, 56, 94
- protection xv
 - See also* data set protection
 - checkpoint 101
 - data set accessibility 97
 - volume 100
 - volume accessibility 96

R

RACF (Resource Access Control Facility)

- IBM standard label
 - checking the next volume 52
 - creating volume labels 42
 - EOV on output 60
 - IEC.TAPERING profile 44
 - opening input data set 45, 49
 - opening output data set 54, 58
 - password checking 48, 57, 58
 - processing on new volumes 60
 - protection and data set name on existing label 57
 - writing data set header labels 58
- ISO/ANSI standard label
 - accessibility 73, 79
 - IEC.TAPERING profile 96
 - opening input data set 103, 108
 - opening output data set 112, 115
 - processing on the new volume 94, 116
 - protection with existing label 94

RACF (Resource Access Control Facility) *(continued)*

- ISO/ANSI/ standard label
 - creating volume labels 92
 - multiple volumes protected by 8
 - nonstandard label
 - opening output data set 125, 126
 - protecting BLP access 12
 - protecting tape volumes 2
 - tape volume defined 1
- RACHECK installation exit 101
- RDBACK parameter
 - description 49
 - IBM standard label 46, 52
 - ISO/ANSI standard label 105, 108, 118
 - ISO/ANSI/ standard label 104, 116
 - no label 124
 - restriction with data conversion 14
- read backward 49
 - See also* RDBACK parameter
- receive step 7
- record
 - format
 - IBM standard label 34
 - ISO/ANSI standard label 83
 - length
 - ISO/ANSI standard data set label 84
- reel label description 139
- REFDD attribute 6
- reflective strip
 - function 15
 - writing beyond
 - IBM standard label 60, 117
- relative generation number 6
- REREAD parameter 10
- Resource Access Control Facility xv
 - See also* RACF
- restart routine
 - from a checkpoint 127
 - function 9
 - IBM standard label 61
 - no label 127
- retention period xv
 - See also* expiration date
- reverse
 - merging DCB 4
 - read xv
 - See also* RDBACK
- reverse merge of information into the JFCB
 - reverse merging information into the JFCB 4
- routine
 - processing 8
- RPG (Report Program Generator) 137

S

- scratch tape
 - IBM standard label 24, 54
 - ISO/ANSI standard label 73, 93, 115
 - no label 126
- security xv
 - status code
 - IBM data set label 1 30
 - ISO/ANSI standard data set label 1 79
 - verification, Open/EOV user exit 47, 56
- SL subparameter 2, 17
- SMS (Storage Management Subsystem)
 - data set 5
- sort/merge program 137
 - See also* DFSORT
- spanned record 101
- specific volume 7
- standard label xv
 - See also* ANSI or IBM standard label
- standard volume label xv
 - See also* volume label
- SUL subparameter 2, 17
- system
 - bypassing label processing 11
 - code
 - IBM standard label 31
 - ISO/ANSI standard data set label 1 80
 - generation
 - 7-track density 13, 15
 - output
 - tape, SYSOUT 55
 - processing 11

T

- tape
 - cartridge
 - density 15
 - high speed search 55
 - tape recording technique 15
 - characteristics 12, 15
 - configuration data base 131, 135
 - created by another system 129
 - DASD advantages 5
 - data set allocation 3
 - disposition 10
 - format 2
 - label xv
 - See also* label, tape
 - layout 2
 - library 21, 24, 48, 70
 - logical write-protection 44, 95
 - management installation exits 43
 - mark
 - no label 11

- tape (*continued*)
 - moving data sets to DASD 5
 - processing 15
 - RACF 2
 - recording technique 15, 36, 85
 - reel label 139
 - reposition routine 119
 - unit 11, 15
 - unlabeled 130
- tape layout 1
- tapemark
 - description 15
 - IBM ISO/ANSI 70
 - IBM standard label 21
 - ISO/ANSI standard label 103
 - no label 123, 124
- track density 13, 15
- trailer label
 - data set definition 20
 - IBM standard 20
 - deferred processing 41
 - description 1
 - format 25, 32
 - IBM user label exit 59
 - ISO/ANSI standard
 - deferred processing 108
 - defined 69
 - format 74, 86
 - nonstandard 13
 - user 21
 - writing data set 59
 - writing user 59
- trailer labels
 - nonstandard 15
- TRTCH parameter 14

U

- UCBDMCT xv
 - See also* mount switch
- UCBDMCT mount switch
 - IBM standard label 45, 54
 - ISO/ANSI standard label 104
- unit exception bit 15
- unlabeled tape
 - component support 137
 - processing 121, 128, 130
 - replaced by labeled tape 11
 - specification 1
 - user identification xv
 - See also* owner identification
 - volume layout 123
- user header label 49
- user label
 - IBM standard
 - component support 137
 - deferred processing 41

- user label (*continued*)
 - IBM standard (*continued*)
 - format 38, 39
 - location on volume 17, 19
 - processing 49
 - processing requirements 21
 - ISO/ANSI standard
 - format 87
 - location on volume 63, 66
 - processing 108, 109
 - writing 115, 116
 - ISO/ANSI/ standard
 - format 88
 - writing 58
- utility program 137
 - See also* IEHINITT program

V

- variable-length record 14, 15
- verification 43, 94, 119
 - See also* volume label verification
- Version 1 xv
 - See also* ANSI standard label
- Version 3 xv
 - See also* ANSI standard label
- version number
 - generation data group 78
 - IBM standard label 29
 - ISO/ANSI standard data set label 1 78
- version numbers
 - description 5
- VOL1 label xv
 - See also* volume label
- volume
 - accessibility 96
 - created by another system 129
 - disposition 10
 - identifier 73
 - initializing 99
 - organization
 - basic layout 2
 - IBM standard label 17, 19
 - ISO/ANSI standard label 64, 66
 - unlabeled tape 121, 124
 - switching
 - IBM standard label 50, 52
 - ISO/ANS standard label 58, 115
 - ISO/ANSI standard label 109
 - no label 124, 125
 - provisions in EOVS routines 9
 - tape 2
 - verification routines 119
- volume label
 - description 1
 - editor routine
 - module name 120

- volume label (*continued*)
 - editor routines 43, 94, 119
 - IBM standard
 - creation 41
 - defined 20
 - format 22, 25
 - processing 40, 45, 52, 53, 58
 - ISO/ANSI standard
 - format 74
 - ISO/ANSI standard
 - creation 68, 92
 - defined 68
 - format 71
 - processing 91, 112
 - verification
 - IBM standard 43
 - ISO/ANSI 94
 - nonstandard 119
- VOLUME parameter 7
- volume sequence number
 - IBM standard label
 - location in header label 28
 - volume switching 50
 - ISO/ANSI standard label
 - volume switching 109
- volume serial number
 - assigned
 - by system 126
 - in JCL statement 24, 25
 - through IEHINITT 24, 25
 - external label 139
 - IBM standard label
 - description 45
 - input processing 45
 - output processing 54
 - switching volumes 50
 - ISO/ANSI standard label
 - description 73
 - input processing 104
 - output processing 113
 - switching volumes 109
 - no label 124, 126
- VSE 9, 32, 121

W

- writing
 - data set
 - header label 58
 - trailer label 59, 60
 - user
 - header label 58
 - trailer label 59, 61
- WTOR installation exit
 - defaults 99

Readers' Comments — We'd Like to Hear from You

DFSMS/MVS Version 1 Release 5

Using Magnetic Tapes

Publication No. SC26-4923-02

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

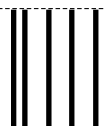
Phone No.



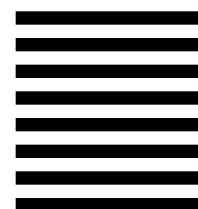
Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
RCF Processing Department
G26/M86/050
5600 Cottle Road
SAN JOSE CA 95193-0001



Fold and Tape

Please do not staple

Fold and Tape



File Number: S370/S390-40
Program Number: 5695-DF1
5647-A01



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC26-4923-02



Spine information:



DFSMS/MVS Version 1 Release 5

Using Magnetic Tapes