IBM

SC26-4511-1

# MVS/ESA
# Magnetic Tape Labels and File Structure Administration

Version 3 Release 1

# MVS/ESA
# Magnetic Tape Labels and File Structure Administration

SC26-4511-1

Version 3 Release 1

**Second Edition (June 1989)**

This edition replaces and makes obsolete the previous edition, SC26-4511-0.

This edition applies to Version 3 Release 1 of MVS/DFP™, Program Number 5665-XA3, and to any subsequent releases until otherwise indicated in new editions or technical newsletters.

The changes for this edition are summarized under "Summary of Changes" following the table of contents. Specific changes are indicated by a vertical bar to the left of the change. A vertical bar to the left of a figure caption indicates that the figure has changed. Editorial changes that have no technical significance are not noted.

Changes are made periodically to this publication; before using this publication in connection with the operation of IBM systems, consult the latest *IBM System/370, 30xx, 4300, and 9370 Processors Bibliography*, GC20-0001, for the editions that are applicable and current.

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 licensed program in this publication is not intended to state or imply that only IBM's program may be used. Any functionally equivalent program may be used instead.

Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality. If you request publications from the address given below, your order will be delayed because publications are not stocked there.

A Reader's Comment Form is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Department J57, P.O. Box 49023, San Jose, California, U.S.A. 95161-9023. IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

# Trademarks

The following names have been adopted by IBM for trademark use and are used throughout this publication:

**ESA/370™**

**MVS/DFP™**

**MVS/ESA™**

**MVS/XA™**

# Contents


**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 . . . . . 4
Cataloged Data Sets . . . . . 5
Generation Data Groups . . . . . 5
Concatenated Data Sets . . . . . 6
Passed Data Sets . . . . . 7
Multiple Volumes and Multiple Data Sets . . . . . 7
SMS-Managed Data Sets . . . . . 8
Processing Methods and Routines . . . . . 8
Data Management Routines . . . . . 8
Checkpoint/Restart . . . . . 9
Automatic Volume Recognition . . . . . 9
Tape Disposition . . . . . 9
Tape Characteristics . . . . . 10
Nine-Track Tapes . . . . . 10
Nine-Track Dual-Density Feature . . . . . 10
Seven-Track Tapes . . . . . 11
Eighteen-Track Tapes . . . . . 13
Tape Marks . . . . . 13
Beginning and End of Tape . . . . . 13

**Chapter 2. IBM Standard Labels** . . . . . 15
Label Definitions and Organization . . . . . 15
Volume Label . . . . . 18
Data Set Header Labels . . . . . 18
User Header Labels . . . . . 18
Data Set Trailer Labels . . . . . 18
User Trailer Labels . . . . . 18
Additional Labels . . . . . 19
Tape Marks . . . . . 19
IBM Standard Label Processing . . . . . 19
RACF Protection . . . . . 21
Opening an Input Data Set . . . . . 21
Volume Serial Number . . . . . 21
Positioning the Volume to the Data Set . . . . . 22
Data Set Name . . . . . 23
Open/EOV User Exit for Security Verification . . . . . 24
Expiration Date . . . . . 24
Password Protection . . . . . 24
Block Count . . . . . 24
Data Set Characteristics . . . . . 24
User Header Labels . . . . . 25
Read Backward . . . . . 25
End-of-Data or End-of-Volume on Input . . . . . 25
Block Count . . . . . 25
The EOV2/EOF2 Label . . . . . 26
User Trailer Labels . . . . . 26
Determining Volume Switch . . . . . 26

Checking the Next Volume . . . . . 28
Closing an Input Data Set . . . . . 29
Creating a Volume Label . . . . . 29
Opening an Output Data Set . . . . . 30
Volume Serial Number . . . . . 31
Positioning the Volume to the Data Set . . . . . 31
Open/EOV User Exit for Security Verification . . . . . 33
Expiration Date on Existing Label . . . . . 33
Password Protection and Data Set Name on Existing Label . . . . . 34
Writing Data Set Header Labels . . . . . 35
Writing User Header Labels . . . . . 35
Permanent I/O Error . . . . . 35
End-of-Volume on Output . . . . . 35
Writing Data Set Trailer Labels . . . . . 36
Writing User Trailer Labels . . . . . 36
Labels on New Volume . . . . . 36
RACF Processing on the New Volume . . . . . 36
Special End-of-Volume Conditions . . . . . 36
Closing an Output Data Set . . . . . 37
Writing Data Set Trailer Labels . . . . . 37
Writing User Trailer Labels . . . . . 37
IBM 3480 Tape Processing . . . . . 37
Restarting from a Checkpoint . . . . . 37
Format of the IBM Standard Volume Label (VOL1) . . . . . 39
Format of IBM Standard Data Set Label 1 (HDR1/EOV1/EOF1) . . . . . 43
Format of IBM Standard Data Set Label 2 (HDR2/EOV2/EOF2) . . . . . 49
Format of User Label (UHL1-UHL8, UTL1-UTL8) . . . . . 55

**Chapter 3. ISO/ANSI/FIPS Labels** . . . . . 57
Summary of ISO/ANSI/FIPS Version 3 Installation and Use Characteristics . 58
System Generation Requirements . . . . . 58
ISCII/ASCII Translation . . . . . 58
Volume Initialization . . . . . 58
ISO/ANSI/FIPS Version 3 Installation Exits . . . . . 58
RACHECK (RACF) Installation Exits . . . . . 59
Volume Protection . . . . . 59
Data Set Protection . . . . . 59
Label Validation . . . . . 59
Generation Data Groups . . . . . 59
Spanned Records . . . . . 60
Block Padding . . . . . 60
Checkpoints . . . . . 60
Compatibility with Non-Version 3 Volumes . . . . . 60
Processing Differences between Version 1 and Version 3 . . . . . 60
Processing Version 3 Volumes on Earlier MVS Systems . . . . . 61
Label Definition and Organization . . . . . 61
Volume Label . . . . . 65
Data Set Header Labels . . . . . 65
Data Set Trailer Labels . . . . . 66
Tape Marks . . . . . 66
Protecting Data . . . . . 67
Volume Accessibility . . . . . 67
Data Set Accessibility . . . . . 68
Overview of ISO/ANSI/FIPS Label Processing . . . . . 70
Opening an Input Data Set . . . . . 71

Volume Serial Number . . . . . 72
Positioning the Volume to the Data Set . . . . . 73
Processing the HDR1 Label . . . . . 74
Processing the HDR2 Label . . . . . 76
Processing User Header Labels . . . . . 76
Read Backward . . . . . 76
End-of-Data or End-of-Volume on Input . . . . . 76
Verifying Block Count . . . . . 77
Determining Volume Switch . . . . . 78
Checking the Next Volume . . . . . 79
Closing an Input Data Set . . . . . 80
Creating a Volume Label . . . . . 80
Opening an Output Data Set . . . . . 82
Volume Serial Number . . . . . 83
Positioning the Volume to the Data Set . . . . . 83
Expiration Date on Existing Label . . . . . 84
Writing Data Set Header Labels . . . . . 84
Writing User Header Labels . . . . . 85
Permanent I/O Error . . . . . 85
End-of-Volume on Output . . . . . 85
Writing Data Set Trailer Labels . . . . . 86
Writing User Trailer Labels . . . . . 86
Labels on New Volume . . . . . 86
RACF Processing on the New Volume . . . . . 86
Special End-of-Volume Conditions . . . . . 87
Closing an Output Data Set . . . . . 87
Writing Data Set Trailer Labels . . . . . 87
Writing User Trailer Labels . . . . . 87
IBM 3480 Tape Processing . . . . . 87
Checkpoint/Restart Not Allowed . . . . . 88
Format of the Version 3 Volume Label (VOL1) . . . . . 88
Format of the Version 3 Data Set Label 1 (HDR1/EOV1/EOF1) . . . . . 91
Format of the Version 3 Data Set Label 2 (HDR2/EOV2/EOF2) . . . . . 97
Format of Version 3 User Header and Trailer Labels (UHL/UTL) . . . . . 103
Other File Header Labels . . . . . 104

**Chapter 4. Nonstandard Labels** . . . . . 105

**Chapter 5. Unlabeled Tapes** . . . . . 107
Bypass Label Processing (BLP) Option . . . . . 107
Opening an Input Data Set . . . . . 107
Positioning the Volume to the Data Set . . . . . 108
Read Backward . . . . . 110
End-of-Data or End-of-Volume on Input . . . . . 110
Determining Volume Switch . . . . . 110
Opening an Output Data Set . . . . . 111
Bypass Label Processing (BLP) Option . . . . . 112
Volume Serial Number . . . . . 113
Positioning the Volume to the Data Set . . . . . 113
End-of-Volume on Output . . . . . 114
Closing an Output Data Set . . . . . 114
Restarting from a Checkpoint . . . . . 114

**Chapter 6. Volume Label Verification and Volume Label Editor Routines** 115

**Chapter 7. Using Tape Volumes Created by Other Systems** . . . . . 117
IBM Standard Labels . . . . . 117
Nonstandard Labels . . . . . 118
Unlabeled Tapes . . . . . 118

**Appendix A. Component Considerations** . . . . . 119

**Appendix B. External Labels** . . . . . 121
Reel Label . . . . . 121
Contents Label . . . . . 121
Checkpoint Security Label . . . . . 122

**Appendix C. Restart Work Areas** . . . . . 123
Table Entry . . . . . 123
Work and Control Block Area . . . . . 126

**Appendix D. ISO/ANSI/FIPS Version 3 Installation Exits** . . . . . 131

**Appendix E. Equivalent ISCII/ASCII, EBCDIC, and Hollerith Codes** . . . . 133
ISCII/ASCII 7-Bit Code . . . . . 133
ISCII/ASCII 8-Bit Code . . . . . 133
EBCDIC to Hollerith and ISCII/ASCII . . . . . 133
ISCII/ASCII to EBCDIC and Hollerith . . . . . 137
Translation Irregularities . . . . . 140

**Abbreviations** . . . . . 141

**Glossary** . . . . . 145

**Index** . . . . . 151

# Summary of Changes

## Second Edition, June 1989

### New Programming Support for Release 1

DATACLAS and the JCL keyword LIKE can be specified for tape data sets. DATACLAS and LIKE= can be used in place of a model data set label for allocating generation data sets. Information on using DATACLAS and LIKE has been added to Chapter 1, "Introduction to Tape Processing."

### New Device Support

Support has been added for the IBM 3424 Magnetic Tape Unit. (The IBM 3424 Magnetic Tape Unit is available only in Brazil, S.A.)

### Service Changes

Minor technical and editorial changes have been made.

## First Edition, December 1988

### New Programming Support for Release 1

Information has been added to Chapter 1, "Introduction to Tape Processing," to describe "automatic concatenation" support of tape and DASD data sets with "like" characteristics.

Considerations for SMS-managed data sets have been added to Chapter 1.

Tape data sets can be moved to DASD to take advantage of the Storage Management Subsystem's capabilities.

### Service Changes

Information on the LOOKAHEAD or PARALLEL tape mounting message has been added to Chapter 2, "IBM Standard Labels" and Chapter 3, "ISO/ANSI/FIPS Labels" to reflect any changes.

Other minor technical changes have been made.

MVS/DFP Version 3 publications have new order numbers. Publications listed in the preface reflect these new order numbers.

# Preface

This book is intended to help you understand and use MVS/DFP processing of magnetic tape labels. It contains general-use programming interfaces, which allow you to write programs that use the services of MVS/DFP. However, this book also provides the following type of information, which is explicitly identified where it occurs:

Product-Sensitive Programming Interface

Installation exits and other product-sensitive interfaces are provided to allow the customer installation to perform tasks such as product tailoring, monitoring, modification, or diagnosis. They are dependent on the detailed design or implementation of the product. Such interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces may need to be changed in order to run with new product releases or versions, or as a result of service.

End of Product-Sensitive Programming Interface

## 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 |
|---|---|
| *MVS/ESA Data Administration Guide* | SC26-4505 |
| *MVS/ESA JCL User's Guide* | GC28-1830 |

## Related Publications

The *MVS/ESA Data Facility Product Version 3: Master Index*, GC26-4512, contains both an index to the MVS/DFP library and a summary of the changes made to the library. You can use it to:

- Find information in other MVS/DFP publications
- Determine how new programming support changes information in the MVS/DFP library
- Determine which MVS/DFP publications have been changed.

In addition, the following publications may be helpful:

| Publication Title | Order Number |
|---|---|
| *MVS Storage Management Library* | SBOF-1241 |

## Referenced Publications

Within the text, references are made to the publications listed below:

| Short Title | Publication Title | Order Number |
|---|---|---|
| Checkpoint/Restart User's Guide | *MVS/ESA Checkpoint/Restart User's Guide* | SC26-4503 |
| Data Administration Guide | *MVS/ESA Data Administration Guide* | SC26-4505 |
| DFP: Customization | *MVS/ESA Data Facility Product Version 3: Customization* | SC26-4504 |
| DFP: General Information | *MVS/ESA Data Facility Product Version 3: General Information* | GC26-4507 |
| Initialization and Tuning | *MVS/ESA System Programming Library: Initialization and Tuning* | GC28-1828 |
| JCL User's Guide | *MVS/ESA JCL User's Guide* | GC28-1830 |
| JES2 Initialization and Tuning | *MVS/ESA System Programming Library: JES2 Initialization and Tuning* | SC28-1038 |
| JES3 Initialization and Tuning | *MVS/ESA System Programming Library: JES3 Initialization and Tuning* | SC23-0073 |
| RACF General Information | *Resource Access Control Facility (RACF) General Information* | GC28-0722 |
| System—Data Administration | *MVS/ESA System—Data Administration* | SC26-4515 |
| System Generation | *MVS/ESA System Generation* | GC28-1825 |
| System Modifications | *MVS/ESA System Programming Library: System Modifications* | GC28-1831 |
| System Reference | *MVS/ESA Diagnosis: System Reference* | LY28-1011 |
| User Exits | *MVS/ESA System Programming Library: User Exits* | GC28-1836 |
| Using Dumps and Traces | *MVS/ESA Diagnosis: Using Dumps and Traces* | LY28-1843 |
| Utilities | *MVS/ESA Data Administration: Utilities* | SC26-4516 |

| Short Title | Publication Title | Order Number |
|---|---|---|
| Magnetic Tape Subsystem User's Guide | *IBM 3480 Magnetic Tape Subsystem: User's Guide* | GC35-0099 |

## Magnetic Tape Label Standards

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
- *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
- *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-Coded Recording*, X3.54-1976
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 32rpmm (800rpi)*, ISO 1863
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 8 and 32rpmm (200 and 800rpi) NRZI, and 63rpmm (1600rpi), Phase Encoded*, ISO 1864
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 63rpmm (1600rpi), Phase Encoded*, ISO 3788

# Chapter 1. Introduction to Tape Processing

Labels are used to identify magnetic tape volumes and the data sets they contain. You can process tape volumes with IBM standard labels, International Organization for Standardization (ISO) or the equivalent American National Standards Institute (ANSI) labels, nonstandard labels, or no labels. Your installation can optionally 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 is the first record on the tape; it identifies the volume and its owner. The data set label groups precede and follow each data set on the volume, and identify and describe the data set.

- 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.

In general, 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 (BCD).

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

Figure 1 on page 2 shows the IBM standard, ISO and ANSI standard, nonstandard, 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 resource access control facility (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.

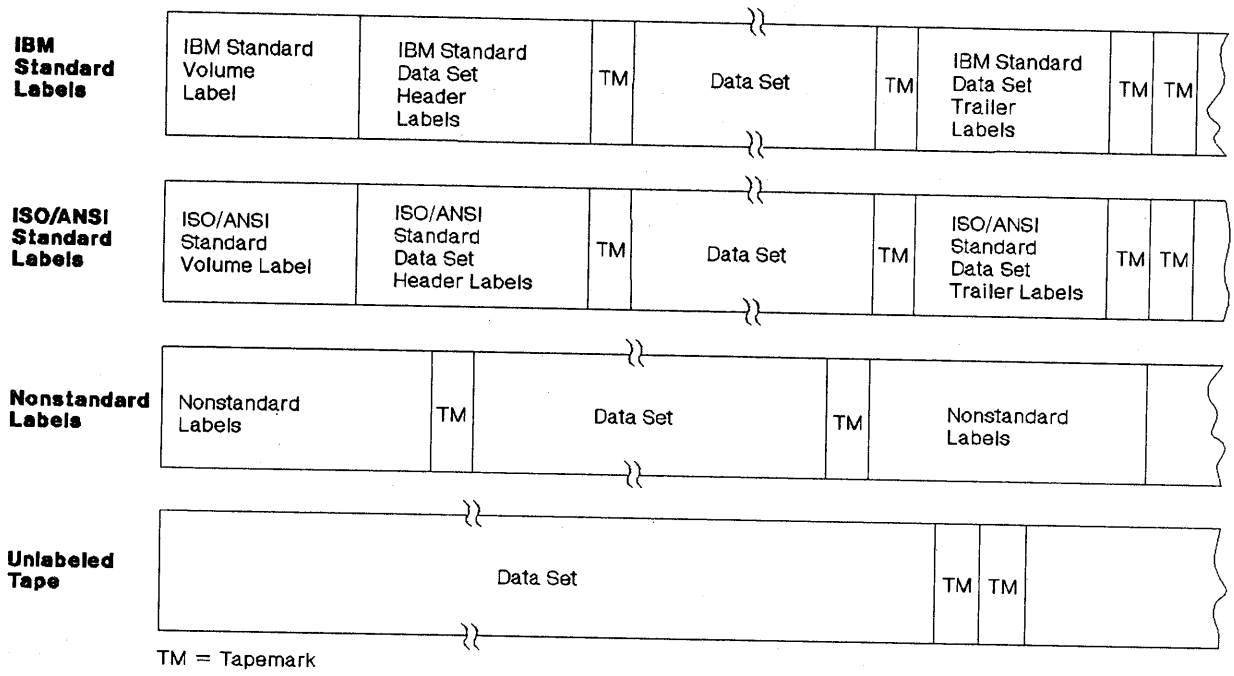



Figure 1. Basic Tape Layouts

## 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/FIPS labels. |
| SUL | Both IBM Standard and user header or trailer labels. |
| AUL | Both ISO/ANSI/FIPS and user header or trailer labels. |
| NSL | Nonstandard labels. |
| NL | No labels. |
| BLP | Bypass label processing (BLP). May be used when a data set having no labels is to be written. 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 support BLP, the data set is treated exactly as if NL had been specified. BLP is an option specified when you initialize Job Entry Subsystem (JES). |
| LTM | Bypass a leading tape mark, 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.

Only SL, AL, SUL, and AUL tape volumes may be protected with the RACF. 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, SUL, AUL, and NSL tape volumes. For NSL tapes, the first data set on the first volume is not a criterion for a valid PROTECT specification.

**Note:** Beginning with RACF 1.7, a volume whose label specification is NL or BLP can be protected; that is, NL and BLP volumes can be protected by RACF commands or by specifying PROTECT=YES on the DD statement.

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 or OUTIN is specified as the processing method in the OPEN macro instruction, the LABEL parameter can be used to 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 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 will be 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. If not otherwise specified, the expiration date is recorded as zeros (allowing the data set to be overwritten immediately), and security (password) protection is not provided.

**Note:** Beginning with RACF 1.7, RACF uses this LABEL parameter value to determine security expiration.

## 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 may use a data class to specify all of your data set's attributes such as record length and record format, except data set name and disposition. You can 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 *Storage Administration Reference* for more information on data class.

**Note:** Tape data sets cannot be SMS-managed.

### Allocating Tape Data Sets

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

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

## Completing the Data Control Block

The information contained in the DD statement is read by the operating system and 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.

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

- The DCB macro instruction in the processing program. The DCB macro instruction is used to construct a DCB and to provide information about the data set.
- The DD statement in the input stream (recorded in the JFCB).
- 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 **7** are bypassed if the tapes have nonstandard labels or no labels.

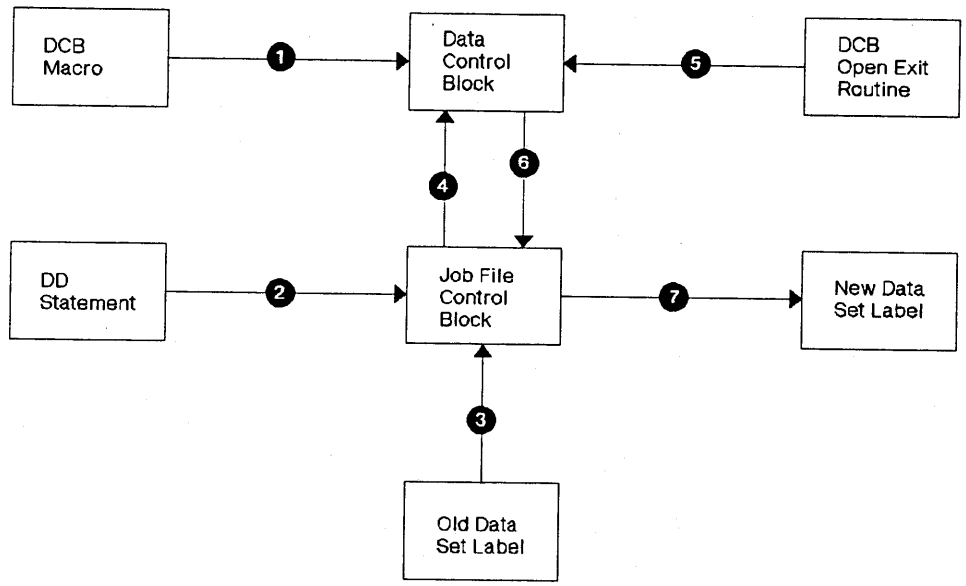



Figure 2. Sources and Sequence for Completing the Data Control Block

*Forward Merge (Steps 3 and 4):* 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, in turn, information from the JFCB is merged into vacant fields of the DCB. (Any fields that were already specified by the DCB macro instruction are not changed.) When the forward merge is completed, your processing program can use the DCB open exit routine to modify the DCB. For a description of the DCB open exit routine, see *DFP: Customization.*

*Reverse Merge (Step 6):* 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.

## 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. The system finds the associated information in the catalog, and issues a mount message to the operator.

## 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.

When a generation data group index is established, a related model data set label must be built on the volume that contains the index. This model label may be used to supply uniform attributes for each generation. If you use a relative generation number to specify a new data set, attributes are taken from the model label. You can override the model label attributes with the DCB parameter of the DD statement.

You no longer need to use a model data set label to allocate a new generation data set. You may use DATACLAS and the JCL keyword LIKE= in place of a model data set label in the DD statement. However, model data set labels 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)

If you do not specify a block size in the JCL, the block size defaults to 32760. 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.

An example of allocating a tape generation data set by supplying its DCB attributes through DATACLAS and LIKE is as follows.

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

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

Information on creating and retrieving generation data groups can be found in *DFP: Customization*, *JCL User's Guide*, and *Utilities*.

## Concatenated Data Sets

Through the technique of concatenation, several different 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 multivolume data set. Concatenated data sets cannot be read backward.

Because only one DCB is associated with all the concatenated data sets, you must inform data management if the data sets have unlike allocation characteristics that do not match (such as device type, block length, and record format). To do this, your processing program must set a switch in the DCB, as explained in *Data Administration Guide*.

When standard or ANSI labels, system managed buffers, and the Queued Sequential Access Method (QSAM) are used, the data set unlike characteristic of "block length" does not apply, and you can concatenate tape data sets in any order of block size. Normal DCB, JFCB, and label merging is performed to obtain tape block size when labels are not present or are bypassed.

When the EXCP, POINT, or CONTROL macro command is not used, you can concatenate tape and DASD data sets if they have like allocation characteristics that match.

More information on concatenated data sets can be found in *Data Administration Guide*.

## 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.

## Multiple Volumes and Multiple Data Sets

You can place a single data set on multiple volumes by coding multiple volume serial numbers in the related DD statement, or by requesting 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 will not be able to retrieve the data set properly through use of the catalog.

You can place multiple data sets on a single volume by coding the same volume serial number on each of the related DD statements, or by using the VOLUME=REF parameter on the DD statements for the second and subsequent data sets. (VOLUME=REF=*.ddname must not be used if the DD statement referred to requests a nonspecific volume.) 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.

You can place multiple data sets on multiple volumes by coding a set of volume serial numbers on each of the related DD statements, or you can use the VOLUME=REF parameter. (VOLUME=REF must not be used if the data set referred to actually used fewer volumes than you specified. VOLUME=REF=*.ddname must not be used if the DD statement referred to requests a nonspecific volume.) If you code a set of volume serial numbers for each of the data sets, the first number must be the serial number of the last volume occupied by the preceding data set.

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 relative position of the data set on the first volume of the group of multiple volumes that it occupies. Data sets are retrieved in an order that differs from the order in which they were written. The specified sequence number must indicate the relative position of the data set on the first volume that it occupies. Therefore, you must not use the catalog to retrieve a data set that is out of order.

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. This will occur automatically as a data set on a

RACF-protected volume is extended onto a new volume that is not RACF protected; the new volume will be defined to RACF as part of the same volume set as the previous volume.

### SMS-Managed Data Sets

If you have DFP 3.1 installed with 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.

For more information on the Storage Management Subsystem, see *MVS/ESA Data Facility Product Version 3: General Information*. Also, *MVS Storage Management Library: Storage Management Subsystem Migration Planning Guide* explains how to move tape data sets to DASD.

## Processing Methods and Routines

The method of processing (INPUT, OUTPUT, RDBACK, INOUT, or OUTIN) is specified by an operand of the OPEN macro instruction. If you do not specify the method, INPUT is assumed.

A data set can be processed as either input or output (INPUT or OUTPUT). 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). For INOUT, the data set is an input data set first and then, without reopening, an output data set. For OUTIN, 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.

### Data Management Routines

The input/output support routines of data management perform the label processing. These routines are open, EOV, and close.

*Opening a Data Set:* The open routine is entered when the processing program issues an OPEN macro instruction. 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.

*End of Data Set or Volume:* The EOV routine is entered when a tape mark is read, when the end of reel (reflective strip) is encountered, or when the processing program issues a force-end-of-volume (FEOV) macro instruction. If you use the execute channel program (EXCP) technique, your processing program must issue an EOV macro instruction to give control to the EOV routine after your program recognizes a tape mark or end of reel. The EOV 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, the EOV 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, EOV gives control to the user's end-of-data routine that is specified in the DCB.

*Closing a Data Set:* The close routine is entered when the processing program issues a CLOSE macro instruction. 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.

## Checkpoint/Restart

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/FIPS labels are not supported by checkpoint/restart. Any ISCII/ASCII tape that is open during a CHKPT macro service will prevent a checkpoint from being taken.

All DOS tapes having either a leading tape mark and/or embedded checkpoint records can be handled by checkpoint/restart, with the exception of DOS 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 to later job steps.

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 on the basis of their serial numbers. AVR is part of the job scheduler (not of data management).

## Tape Disposition

Tape disposition at end of data set or end of volume can be influenced 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 instruction. The OPEN macro instruction controls positioning after an end-of-volume condition (multivolume data sets) unless overridden by the FEOV macro instruction. The CLOSE macro instruction 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 — Specifies that the data set associated with this DCB is to be released for use by another task and that the device on which the data set is mounted can be freed for allocation to another job.

If none of the above are specified, DISP is assumed.

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.

## Tape Characteristics

The following paragraphs describe the data recording characteristics of magnetic tape. The discussion includes density, parity, number of tracks, translation, conversion, tape marks, and so forth, as related to the operating system and to IBM 3400 series Magnetic Tape Units. The error conditions that can result from conflicting tape characteristics are explained under Chapter 6, "Volume Label Verification and Volume Label Editor Routines" on page 115.

The phrase "IBM 3400 Magnetic Tape Units" refers to the IBM 3410, the IBM 3420 Models 3, 4, 5, 6, 7, and 8, the IBM 3422, the IBM 3424[1], the IBM 3430 Magnetic Tape Subsystem, and the IBM 3480 Magnetic Tape Subsystem.

### Nine-Track Tapes

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

### Nine-Track Dual-Density Feature

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 on page 11 shows the DEN parameter codes). If the DEN parameter code is not specified, the greater density is assumed.

[1] The 3424 Magnetic Tape Unit is availalbe only in Brazil, S.A.

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, provided that 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.

| | Recording Density | | |
|---|---|---|---|
| **DEN Value[1]** | **7-Track** | **9-Track** | **18-Track** |
| 1 | 556 | – | N/A |
| 2 | 800 | 800 (NRZI) | N/A |
| 3 | – | 1600 (PE) | N/A |
| 4 | – | 6250 (GCR) | N/A |

NRZI is for Nonreturn-to-zero-inverse mode.

PE is for Phase encoded mode.

GCR is for Group coded recording mode.

[1] If the DEN parameter is not supplied by any source, the highest applicable density is assumed.

Figure 3. DEN Parameter Codes for Specifying Tape Density

## Seven-Track Tapes

The 7-track feature allows a tape unit to read and write 7-track tapes. 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 (BCD) 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 six low-order bits of the characters in your buffer are written on tape; during reading, the two 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 three characters from your buffer as four 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 instruction (Figure 3 on page 11 shows the DEN parameter codes). If not specified, the default value is 800 bpi.

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

If you use densities other than 800 bpi for 7-track system input tapes (SYSIN), system output tapes (SYSOUT), or tapes to be handled by the AVR option, you must establish the particular density for each during the system generation process.

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 parameter are:

| 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 tape 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 tape 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 tape marks should be carefully planned. System recognition of tape marks is ensured only when they are read in the density in which they were written.

## Eighteen-Track Tapes

The operating system supports the IBM 3480 Magnetic Tape Subsystem, which uses 18-track recording. For 3480 tapes, only a single density is available and is used by the system for reading and writing; any density with the DEN parameter is ignored.

ISO and ANSI standards do not include a specification of 18-track magnetic tape for information interchange. Therefore, 18-track recording is not applicable for tapes with ISO or ANSI labels.

## Tape Marks

A data set or label group on tape is usually followed by a tape mark delimiter. A tape mark is a special character written by a control command. (In the figures of this manual, tape mark is represented as "TM.") The tape drive recognizes a tape mark during a read operation and signals a unit exception condition. The condition is displayed by the unit exception bit in the channel status word (CSW), where it is recognized by the operating system. The tape mark 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 CSW.

In contrast to the above, the IBM 3480 Magnetic Tape Subsystem has an internal mechanism that senses the beginning and end of tape. Reflective strips are not used.

# 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 data management routines of the operating system perform IBM standard label processing. If you specify SUL, data management processes both IBM standard labels and IBM standard user labels.

This chapter describes the organization, formats, and contents of IBM standard labels, and explains how they are processed or created.

## Label Definitions and Organization

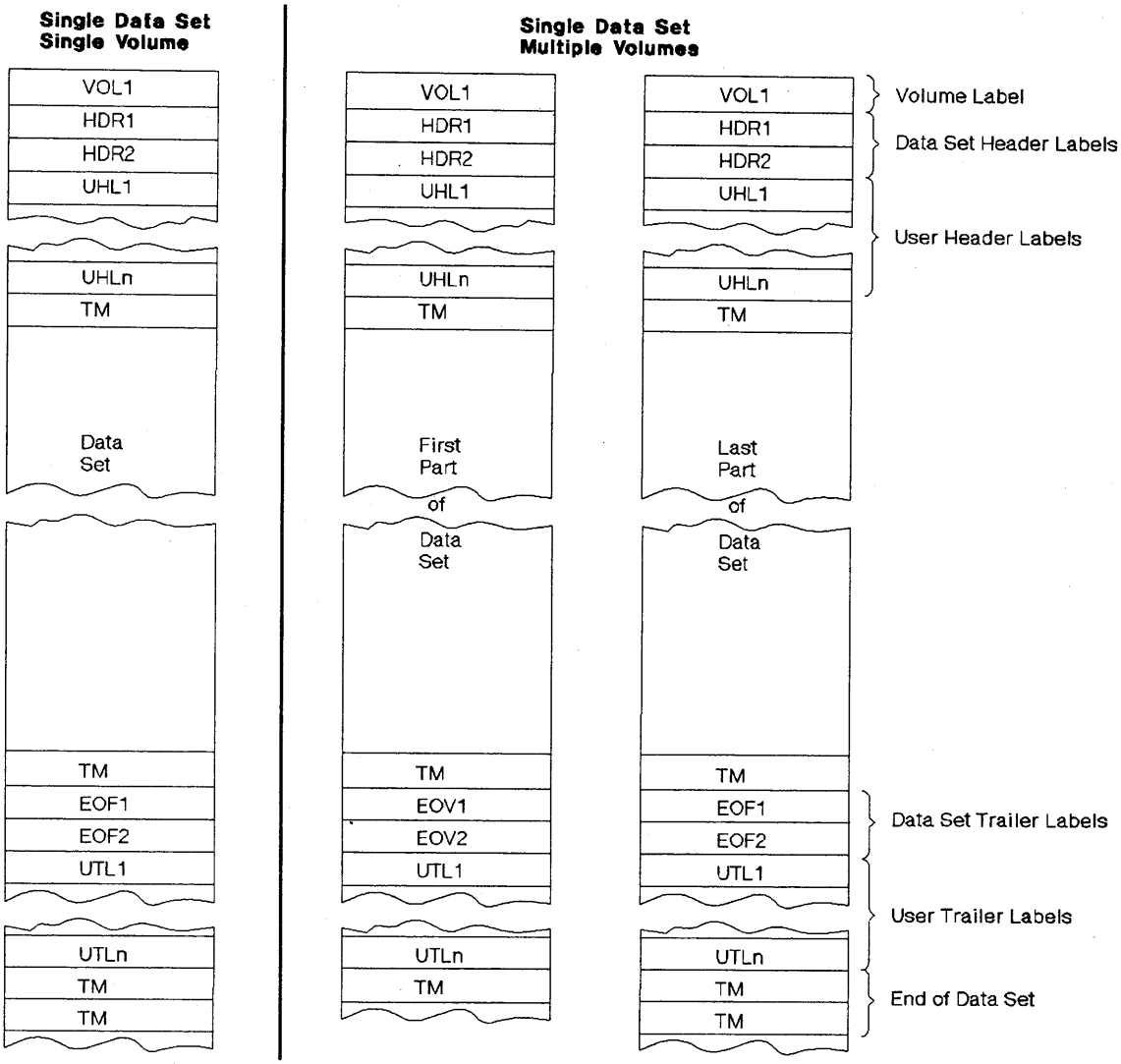
IBM standard labels are 80-character records written in the density specified via JCL. Seven-track tape records are recorded in BCD, even parity, translate on. The first four characters are always used to identify the labels:

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

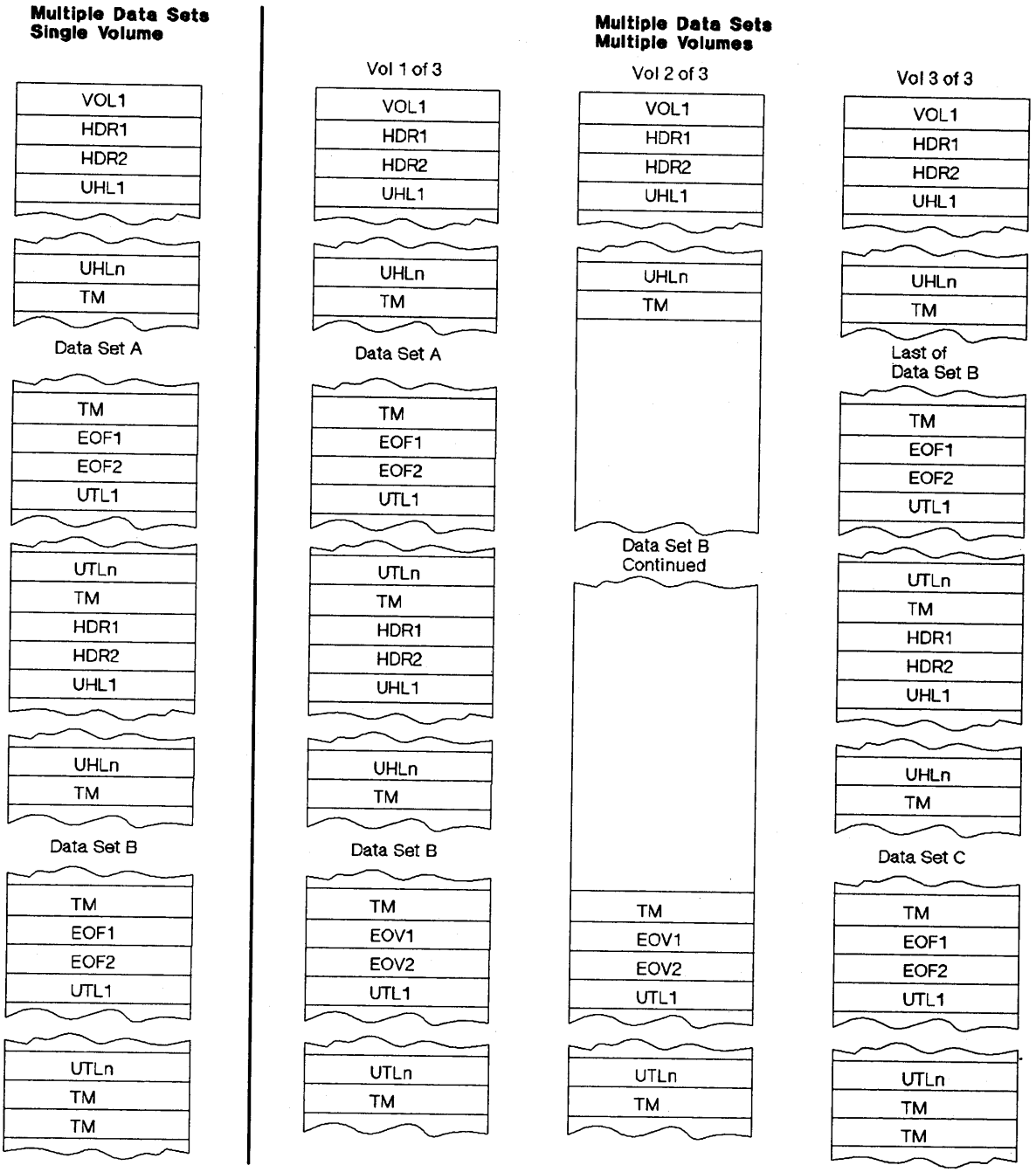
Figure 4 on page 16 and Figure 5 on page 17 show the positions of the labels with various tape volume organizations. A tape with IBM standard labels must contain a volume label and data set labels. User labels are optional.




**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 4. 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 fo 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 5. 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.

The volume label is created by either a utility program or the user's program when the tape is first received at an installation.

## 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 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 data management 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 data management 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 (EXLST) is described in *DFP: Customization.*

## 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 BCD).

These labels are created automatically by data management 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 data management 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 *DFP: Customization*.

### 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.

### Tape Marks

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

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

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

## IBM Standard Label Processing

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

These processing functions are summarized in Figure 6 on page 20. The table shows the specific labels that are processed for each function and which routines perform the functions. The summary in Figure 6 is the basis for the discussions of label processing that follow.

| Processing | Volume Label VOL1 | Header Labels [1] | | | Trailer Labels [1] | | |
|---|---|---|---|---|---|---|---|
| | | HDR1 | HDR2 | UHL1–8 | EOF1 or EOV1 | EOF2 or EOV2 | 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 | not read | not read | not read | not read | Open [3] |
| Writes new labels on output tape. | Open or user [4] | Open | Open | Open | Close or EOV | Close or EOV | Close or EOV |
| **Second or subsequent volumes:** [5] | | | | | | | |
| Checks labels on input tape. | EOV | EOV | bypassed | EOV | EOV | bypassed | EOV |
| Checks 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 EOV | Close or EOV | Close or EOV |

**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. (Data sets with like characteristics can be processed correctly using the same data control block (DCB), input/output block (IOB), and channel program. Any exception in processing makes the data sets unlike.)

[3] If DISP=MOD is specified on the DD statement, 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 IEHINITT utility program 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 with like characteristics.

Figure 6. IBM Standard Label Processing by Data Management Routines

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 instruction 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 was 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.

## RACF Protection

ALTER access authority is required to create or destroy a tape volume label. UPDATE access authority is required when opening a RACF-defined tape volume for output (including INPUT or OUTIN specification). READ access authority is required when opening a RACF-defined tape volume for input. For an overview of RACF protection for tape volumes, see *RACF General Information.*

## 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, data management 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 Chapter 6, "Volume Label Verification and Volume Label Editor Routines" on page 115.

### Volume Serial Number

Data management 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, or RDBACK.

If the volume serial number is correct, data management 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, data management rejects the tape and issues another mount message.

After the volume is mounted and verified and if the system-wide RACF tape protection option has been specified, RACF access authorization to the volume is checked. READ authorization is checked if open for INPUT or RDBACK. If open for INOUT, UPDATE authorization is checked. If the volume is not defined or if the user is authorized to the volume, processing continues; otherwise, the program is abnormally terminated.

## Positioning the Volume to the Data Set

When the volume is mounted and verified, data management 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 reel, and the header label group immediately follows the volume label.

To retrieve a data set when more than one data set is on a single reel of tape, 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 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, your task will be abnormally terminated or, if the volume ends with EOV 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, data management assumes that the data set is the first in sequence on the volume.

To position the tape, data management 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 or OUTPUT 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 EOV 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 or OUTPUT DISP=MOD to the end of the data set, 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=DSNAME 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 tape mark.

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 EOV 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 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.

Only one data set on a tape volume may be open at any given time. An attempt to begin processing of 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, data management checks the label identification. If the identifier HDR1 is not found, processing is abnormally terminated.

## Data Set Name

To ensure that the correct data set is being opened, data management 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).

If the comparison shows an incorrect data set name, processing is abnormally terminated.

## 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 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 *DFP: Customization*.

## 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, data management 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.)

## Password Protection

If the volume is RACF protected and authorization to access the volume was checked when the volume was verified, password checking will be bypassed for any data set on the volume. Password protection is not recommended for any volume data set; RACF protection is preferred.

## Block Count

The block count shown in the HDR1 label is always 0 (EBCDIC or BCD). 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.

## Data Set Characteristics

The HDR2 label immediately follows the HDR1 label. Data management 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 (7-track tape)
- Type of control characters

The above information is obtained from the label and recorded in the JFCB and the data control block, provided 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. This merging process is explained and illustrated in the introduction to this manual.

Unless user header labels are to be processed, data management positions the tape past the tape mark 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, data management positions the tape past the tape mark immediately after processing the HDR2 label.

### Read Backward

For the read backward (RDBACK) processing method, data management 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, 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

Data management's EOV routine handles both end-of-data-set and end-of-volume conditions on input. These conditions occur when:

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

After encountering a tape mark, data management checks the first 4 bytes of the first trailer label for the identifier EOV1 or EOF1. If neither identifier is found, processing is abnormally terminated. When the FEOV macro instruction is executed, the trailer labels are not checked.

### Block Count

To verify that all records on the input data set on the current volume have been read, data management 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, data management 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 0B (hexadecimal). 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 237 (hexadecimal). |
| 4 | Continue processing. |

If you do not provide the appropriate user exit entry in the DCB exit list, a block count discrepancy will cause processing to abnormally terminate with a completion code of hexadecimal 237. When the FEOV macro instruction is executed, the block count is not verified.

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

## The EOV2/EOF2 Label

Except when it is used as a header label for a read backward operation, data management ignores the second trailer label (EOV2 or EOF2) of an input data set.

## User Trailer Labels

If user trailer labels (UTL1-UTL8) are present on input, data management 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.

## 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. 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. Data management 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 volume sequence number in the VOLUME parameter of the DD statement.

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

the DD statement specifies OPTCD=B, and if an additional volume is available, data management performs volume switching. If OPTCD=B and no volumes are available for switching, data management passes control to the user's end-of-data routine.

To determine whether additional volumes are required, data management maintains a volume sequence number in the data extent block (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), data management 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 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 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 don't want message retention, turn off the message retention attribute. You can accomplish this by using an exit such as IEECVXIT, IEAVMXIT, or other user exit to alter the message descriptor code to type 3. For additional information on these user exits, see *User Exits* and *System Modifications*.

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 DOM (delete operator message) to be initiated against the IEC501E message and will issue 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 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.

If the system-wide RACF tape protection has been specified, RACF access authorization to the volume is checked. READ authorization is required if open for INPUT or RDBACK. If open for INOUT or OUTIN, UPDATE authorization is checked. If a new concatenated data set is not being processed and the open is for INOUT or OUTIN (not INPUT or RDBACK), it will further be verified that, if the new volume is RACF defined, it is defined as part of the same volume set profile as the previous volume.

Processing continues normally if:

- The new volume is not defined to RACF or
- The user is authorized.

For a specific volume request, the program is abnormally terminated if:

- The new volume is defined to RACF and
- The user is not authorized.

For a nonspecific volume request, the operator is asked to mount a new volume.

If the user is READ authorized but not UPDATE authorized, and the data set is open for INPUT or RDBACK, but the tape is not file protected, the volume is demounted and the operator is instructed to remove the write-enable (file protect) ring and remount the tape.

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. (Data sets with like characteristics can be processed using the same DCB, input/output block (IOB), and channel program.) In either case, the user exit for security verification may be taken, as described in "Open/EOV User Exit for Security Verification" on page 24.

- *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. If the data set name is not the same, or the password was not verified for each volume, processing is abnormally terminated.
- *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 in the same manner as when the first

data set was opened, including verification of the password if protection is indicated.

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.

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 or the end-of-volume, or if the FEOV macro instruction is executed, processing of trailer labels is omitted.

## Creating a Volume Label

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

You can use the IBM-supplied IEHINITT utility program to create IBM standard volume labels. IEHINITT initializes 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 tape mark.

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 *Utilities*.

Methods other than the IEHINITT utility program 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 (see Chapter 6, "Volume Label Verification and Volume Label Editor Routines" on page 115) with one that writes volume labels. If you use an editor routine to write the volume label, some data or a tape mark should already exist on the tape; otherwise, data management reads through the entire reel of blank tape looking for a label.

Except for the IBM 3480 Tape Subsystem, the VOL1 label is rewritten by the open or EOV routine if **all** the following conditions are met:

- A density conflict has occurred.
- OUTPUT or OUTIN is specified in the OPEN macro instruction.
- 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.

On an IBM 3480 Magnetic Tape Subsystem the open and EOV modules bypass rewriting a VOL1 label, thus improving performance.

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 (that is, you need not specify a volume serial number in your DD statement) for output tapes and allows the operator to mount any available scratch tape 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 EOV routine creates a volume label (VOL1) and sends a message to the console operator requesting serial number and owner information.

## 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, data management 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 Chapter 6, "Volume Label Verification and Volume Label Editor Routines" on page 115.

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 specification is valid; both the volume sequence number and the file sequence number must be set to one 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 will not attempt validity checking (of the PROTECT=YES specification) and definition of the volume to RACF.

**CAUTION:**
**Doing multiple opens and closes without writing any user data in the area of the end-of-tape reflective marker may result in the header and trailer labels being written past the marker. Access methods detects the markers, but because the creation of empty data sets does not involve access methods, the end-of-tape marker will never be detected. This may cause the tape to run off the end of the reel.**

## Volume Serial Number

You are not required to specify volume serial numbers for output tapes. If none is specified, the mount message directs the operator to mount a scratch tape. Data management obtains the volume serial number from the VOL1 label and records it in the JFCB and the UCB. A user exit is provided to allow you to identify a specific tape volume in place of a nonspecific (scratch) volume. The exit is invoked when open or EOV is to issue a mount request for a volume for which no volume serial number has been specified, and will get control before the mount message is issued. For more information on the "Open/EOV nonspecific tape volume mount" exit, see *DFP: Customization*.

If you choose to specify the volume serial number, data management compares it with the volume serial number shown in the VOL1 label. If the number is correct, data management 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 volume serial number is incorrect, data management 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, or is not a checkpoint/restart volume. Otherwise, data management rejects the tape and issues another mount message. Volumes are requested for mounting in the order specified.

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, or if the tape volume is defined and the user is UPDATE authorized and PROTECT=YES has been specified, processing continues. If the tape volume is defined and either the user is UPDATE authorized and PROTECT=YES has been specified, or the user is not UPDATE authorized, the volume will be rejected if a nonspecific request is made and 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, data management positions the tape to receive the new data set. Usually the new data set will be 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 EOV 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.
- If you do not specify a sequence number, or specify zero, data management assumes that the data set is to be written as the first on the volume.

To position the tape, data management 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. If you attempt to open another data set on the same volume, processing is abnormally terminated. This restriction includes system output (SYSOUT) tapes.

When the tape is positioned to receive the new data set, data management expects to find either an existing HDR1 label or a tape mark. If neither is present, data management 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 EOV labels, another data set cannot be written to follow it.)

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

- When a tape mark immediately follows the VOL1 label. This may occur when the tape is initialized by means other than the IEHINITT utility program (IEHINITT writes a dummy HDR1 label following the VOL1 label). The tape mark 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, data management encounters the second tape mark following the existing EOF trailer label group. The tape mark is overwritten by the new HDR1 label.

If data management 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 3480 Tape Volume

When you want to extend a data set on an IBM 3480 tape volume, either 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 (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 tape mark 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.

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 *IBM 3480 Magnetic Tape Subsystem User's Guide*.

If a JFCB, with the "fast positioning" indicator on, is 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 will position the tape normally and insert 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 tape mark immediately following the last block of user data when opening to extend a data set.

Once you have turned on the "fast positioning" indicator in a JFCB for an OPEN with TYPE=J in a job step, you should 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 *DFP: Customization*.

## Expiration Date on Existing Label

The existing HDR1 label is inspected for the expiration date. If the expiration date has not been reached, the operator is asked to confirm use of the tape or to mount another tape.

If other data sets exist on the same volume, data management 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, data management 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, the tape is accepted for output. Data management does not request a password, and does not check the data set name.

If protection is indicated, data management 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 demount 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 either the IEHINITT utility 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 has been found to be defined to RACF.
- 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, data management 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.

*System—Data Administration* 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 tape marks 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 data management for output, data management 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 each label is explained in the description of label formats. The process of updating the JFCB is explained in the introduction.

The security indicator is set in the HDR1 label even if the volume is RACF defined.

If no user header labels are to be written, data management writes a tape mark 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, data management 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 demount the tape and mount a scratch tape, even if you request 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 will be abnormally terminated.

## End-of-Volume on Output

Data management's EOV routine automatically switches volumes when an end-of-volume condition occurs, that is, when a reflective strip is encountered or when a FEOV macro instruction is executed. This volume switching includes:

- Writing trailer labels on the current volume
- Checking existing labels on the new volume
- Writing header labels on the new volume

When multiple tape units are being used, the EOV routine 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 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 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 don't want message retention, turn off the message retention attribute. You can accomplish this by using an exit such as IEECVXIT, IEAVMXIT, or other user exit to alter the message descriptor code to

type 3. For additional information on these user exits, see *User Exits* and *System Modifications*.

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 DOM (delete operator message) to be initiated against the IEC501E message and will issue 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 EOV1 and EOV2 (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 tape mark is written following the data set trailer labels (instead of two tape marks for end of data). If user trailer labels are to be written, the tape mark 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, data management can write as many as eight user trailer labels (UTL1 to UTL8).

## 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. The user exit for security verification may be taken, as described in "Open/EOV User Exit for Security Verification" on page 33.

When creating 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 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, then the new volume must not be RACF protected. If the above conditions are not met, the new volume will be rejected if a nonspecific request is made, or the program will be abnormally terminated if a specific request is made.

## Special End-of-Volume Conditions

When a reflective strip causes an end-of-volume condition during the writing of data, the EOV routine writes the trailer labels as described above. If the reflective strip is encountered while 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 even though it crosses the reflective strip.

If you add another data set to a tape (multiple data set organization) on which the last existing trailer label group crossed the reflective strip, or on which the new header label group crosses the reflective strip, data management:

- 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, or INOUT) or when no output is written before closing (for OUTPUT or OUTIN), 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 tape marks following the trailer labels. If user labels are to be written, the tape marks follow the user trailer labels. If another data set is added to the tape (multiple data set organization), its HDR1 label overlays the second tape mark.

### 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).

## IBM 3480 Tape Processing

To improve performance when processing labels of standard labeled volumes on an IBM 3480 Tape Subsystem, the open and EOV routines attempt to avoid changing the direction of tape movement.

## 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:

1. Restores applicable control blocks to the conditions that existed when the checkpoint was taken.
2. 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.
3. 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 previously explained for opening an input data set. The data set labels are not reprocessed.

4. 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 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.

## 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 BCD on 7-track tape units.

Figure 7 on page 40 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 following the figure. The processing descriptions refer to the following system control blocks:

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

The location of areas within these control blocks can be found in *Data Areas*.

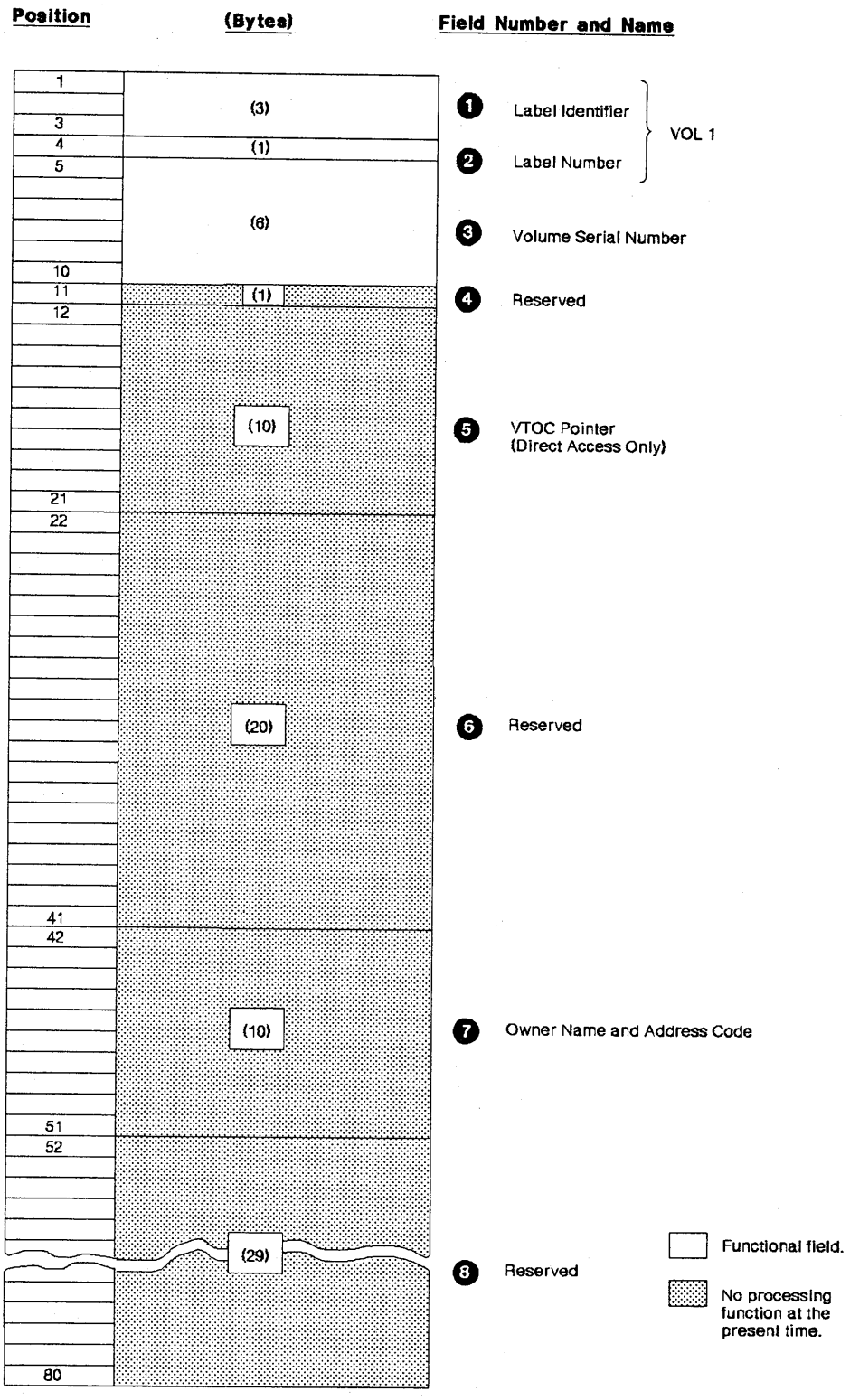



Figure 7. Format of IBM Standard Volume Label

**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 IEHINITT utility program or a user's program.

  In the following situations, open or EOV routines create a standard volume label based on specifications in your DD statement:

  - When you request a standard labeled (SL, SUL) output volume, and an NL or NSL labeled volume is mounted to satisfy your request
  - When you request a standard labeled (SL, SUL) output volume, and a volume that can be overwritten and that is recorded in the wrong density is mounted to satisfy your request

**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 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 IEHINITT to the volume when it enters the system, or that is assigned by the operator when the open or EOV 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 (000001 to 999999), but may be any six alphameric characters. It may be from one to six characters, but, if fewer than six characters, the code must be left-justified, and the remainder is padded with blanks.

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

- *Processing*: The user-specified volume serial number is obtained from the JFCB and recorded in the UCB. Then the number in the UCB is compared 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 IEHINITT utility program can create this label with a volume serial number of up to six 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** (10 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 IEHINITT utility program writes blanks in this field.

**6—Reserved** (20 bytes)

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

**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 IEHINITT utility program writes the text specified by the user, and the open and EOV routines write the text specified by the operator. If the code is less than 10 bytes long, it is left-justified and the remainder of the field is padded with blanks.

**8—Reserved** (29 bytes)

- *Contents*: Reserved for possible future use—should be recorded as blanks.
- *Processing*: Not used or verified. The IEHINITT utility program writes 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. It is recorded in EBCDIC on 9-track tape units, or in BCD on 7-track tape units.

Figure 8 on page 44 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 following the figure. 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)

The location of areas within these control blocks can be found in *Data Areas*.

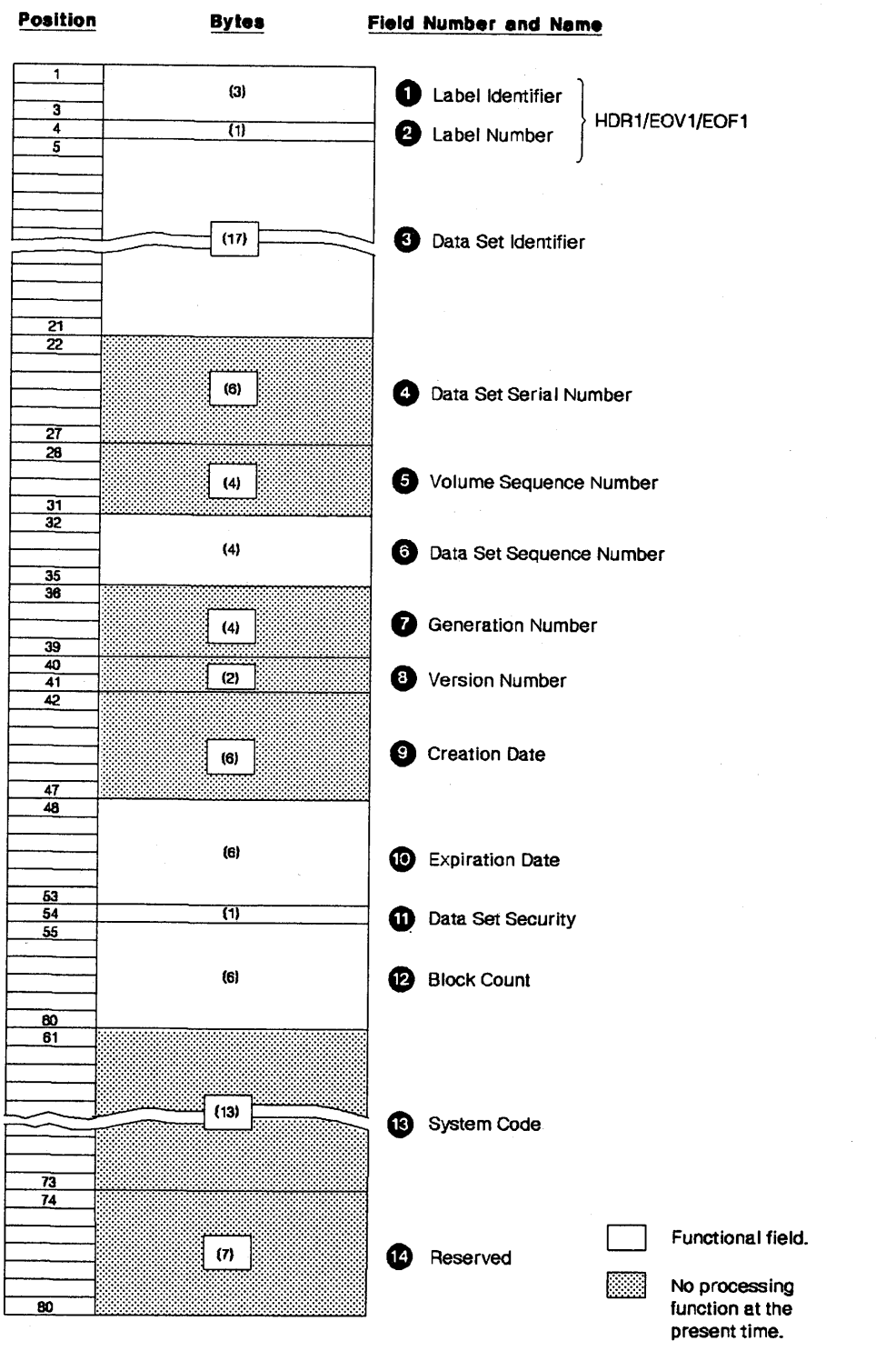



Figure 8. Format of IBM Standard Data Set Label 1

**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*: Data management checks this field to verify that the record is an IBM standard data set label.

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

  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, data management performs the switching. If no volumes are available, data management passes control to the user's end-of-data routine.

  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. *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.

  OS tape input files can accept generation data group members written in DOS format. That is, on input, the operating system can verify the file name field on a DOS tape, excluding the generation and version number as a level of qualification, and verify 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 six alphameric characters, normally numeric (000001 to 999999). The number may be from one to six characters, but, if fewer than six 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 nonalphameric characters are used.
- *Processing*: Not used or verified. 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*: Not used or verified. When creating labels, the open routine writes 0001 in this field; the EOV and close routines obtain the current volume sequence number from the DEB.

**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 EOV routine obtains this number from the logical data set sequence number in the UCB.

**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, data management 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 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, data management checks the JFCB to determine whether 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*: Year and day of the year 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)
```

- *Processing*: Not verified. When data management creates labels, the date is obtained from the JFCB. This is the date the job was initiated for execution, and not necessarily the date the label was created.

**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)
```

- *Processing*: For input, not used or verified. For output, the expiration date in the existing label is compared to the current date shown in the communications vector table (CVT). 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. If any other data sets follow on the same volume, they are considered to expire on the same day.

  When creating labels, data management obtains the expiration date from the JFCB. If you did not specify a retention period or expiration date, the expiration date is recorded as zeros and the data set is considered to have expired.

**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. (Ignored if volume is RACF defined.)

  3 Password protection. Additional identification of the data set is required before it can be written or deleted. (Ignored if volume is RACF defined.)

- *Processing*: For input, data management 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, data management verifies the password furnished by the operator and sets a security indicator in the JFCB.

  For output, data management inspects this field in the existing HDR1 label. If security is specified, the existing data set cannot be overwritten until data management 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 demount 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.

  When the second or greater 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 job is abnormally terminated. If the security levels are equal and indicate no security protection, open processing continues. If security protection is indicated, data management 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 security-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 data management creates labels, the user's request for security is determined from the indicator in the JFCB.

**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 zeros (000000).

- *Processing*: The DCB count 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 will continue or abnormally terminate. If the appropriate user exit entry is not provided, a block count discrepancy causes processing to abnormally terminate.

  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 data management 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 a 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. When creating labels, the code is obtained from the JFCB, where it is always binary zeros.

**14—Reserved** (7 bytes)

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

## 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, or in BCD on 7-track tape units.

Figure 9 on page 50 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)
- Task input/output table (TIOT)
- Unit control block (UCB)

The location of areas within these control blocks can be found in *Data Areas*.

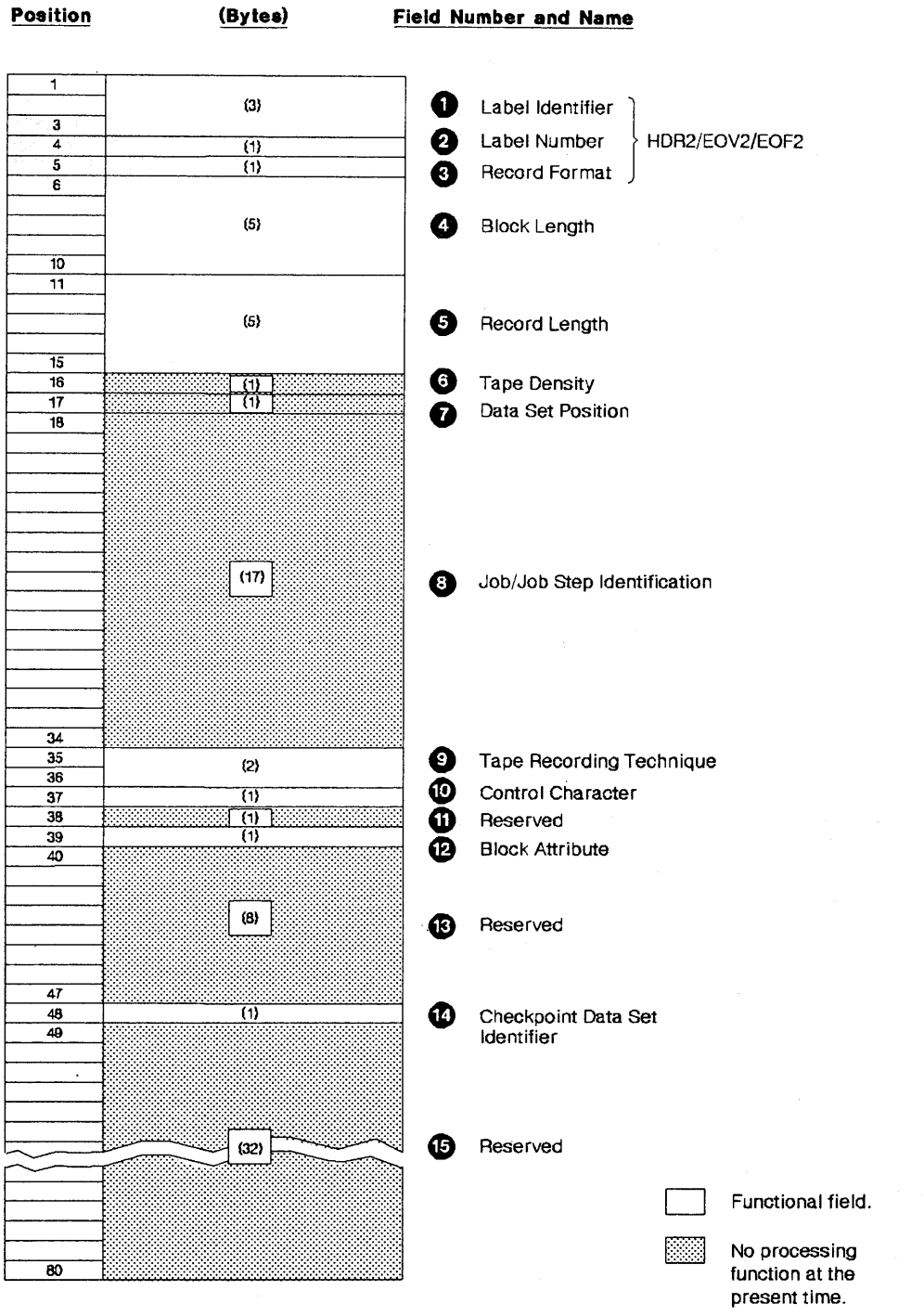



Figure 9. Format of IBM Standard Data Set Label 2

**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*: Data management checks this field to verify that the record is an IBM standard data set label.

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

  If the DD statement specifies OPTCD=B for an input data set, data management 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, data management performs the switching. If no volumes are available, data management passes control to the user's end-of-data routine.

  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 record format is obtained from this label and recorded 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 that this is a merging process in which existing specifications in the JFCB and DCB cannot be overridden.

  When creating labels, a reverse merge follows the forward merge described above. 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 the introduction.

**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 Block length (must be a multiple of the logical record length in Field 5)
  - Format V Maximum block length (including the 4-byte length indicator in the blocks)
  - Format U Maximum block length
- *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 indicator 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 10).

  **Note:** Specifying DEN=0 for a 7-track 3420 will result in 556 bits-per-inch recording, but corresponding messages and tape labels will indicate 200 bits-per-inch recording density. For the 3480 tape, this field contains a blank.
- *Processing*: Not used or verified. When data management 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 EOV routine writes 1. The close routine determines which code to write by comparing the 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 9th byte is a slash (/), and the final 8 bytes contain the name of the job step.
- *Processing*: Not used or verified. When data management creates labels, the names of the job and job step are obtained from the TIOT.

**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

  For other tapes, this field is recorded as blanks. The only technique available for 9-track tape is odd parity and no translation.
- *Processing*: For 7-track tape, 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 ISO/ANSI/FIPS control characters
  - M Contains machine control characters
  - b 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, data management 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'
- b 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** (8 bytes)

- *Contents*: For the 3420, bytes 40 through 42 are reserved, 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 the 3480, bytes 40 through 42 are reserved, bytes 43 through 46 contain the last 4 digits of the serial number of the control unit, and byte 47 contains the device address.

  **Note:** The serial numbers in the header and trailer labels may not be the same if the data set was opened for update or if DDR was used to swap 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 hexadecimal, and inserted into the UCBCTD field in the UCB tape extension.

**14—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 and, 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 will not allow access to the checkpoint data set directly, although it will allow the taking of checkpoints and performing restarts.

**15—Reserved** (32 bytes)

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

# Format of User Label (UHL1-UHL8, 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 10 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 BCD. The contents and processing of each field of the label are described below.

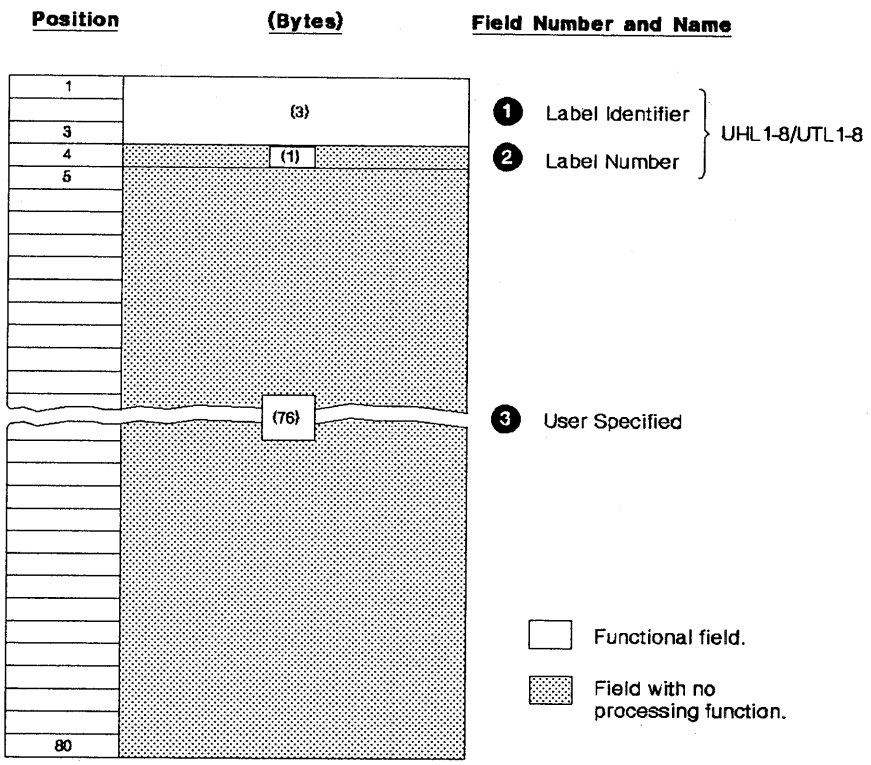



Figure 10. Format of User Label

**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. Data management 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.

# Chapter 3. ISO/ANSI/FIPS 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 as of April 1983:

- ISO 1001-1979, level 4
- ANSI X3.27-1978, level 4
- Federal Information Processing Standard (FIPS) 79

In this manual, the term "Version 3" is used when referring to these standards. The term "Version 1" is used when referring to the previous ISO 1001-1967 and ANSI X3.27-1969 standards. "ISO/ANSI" is used when referring to labels designed according to the specifications of Version 1 standards. "ISO/ANSI/FIPS" refers to Version 3 labels.

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

ISO/ANSI/FIPS support will execute 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, 32rpmm (800rpi)*, ISO 1863
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 8 and 32rpmm (200 and 800rpi) NRZI, and 63rpmm (1600rpi), Phase Encoded*, ISO 1864
- *Information Processing—9-Track, 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange, 63rpmm (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-Coded Recording*, X3.54-1976

Both *Information Processing—9-Track 12.7mm (0.5in) Wide Magnetic Tape for Information Interchange Recorded at 8rpmm (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, and tapes written according to those standards are not supported.

All data on a tape with ISO/ANSI/FIPS labels must be recorded in the 128 characters of basic ISCII/ASCII. Do not use Version 3 volumes to contain raw binary, floating point, or packed decimal data (see "ISCII/ASCII Translation," below). Unlabeled tapes recorded in ISCII/ASCII can be processed, as explained in Chapter 5, "Unlabeled Tapes" on page 107.

# Summary of ISO/ANSI/FIPS Version 3 Installation and Use Characteristics

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

## System Generation Requirements

Operating system support for Version 3 consists of processing labels and translating from ISCII/ASCII to EBCDIC on input and from EBCDIC to ISCII/ASCII on output. This support is included in the system (in module IGC0010C of SYS1.LPALIB) at system generation time by specifying ASCII=INCLUDE in the CTRLPROG macro.

When ISO/ANSI/FIPS support is included in the system, a class extension area is generated for each UCB that represents a tape device. This area is used by ISO/ANSI/FIPS processing. For more information, see *DFP: Customization.*

## ISCII/ASCII Translation

The IBM-supplied translation routine translates ISCII/ASCII 7-bit code, and is designed to support Version 3 standards. The system forces OPTCD=Q during open to cause ISCII/ASCII translation during processing of any volume mounted as LABEL=AL. Note that, if the input data includes raw binary, floating point, or packed decimal fields, any bytes containing values that translate into more than 7 bits are converted to X'1A', resulting in permanent loss of data.

If necessary, you may replace the IBM-supplied translation routine, which is an SVC routine, with an installation-written routine that translates ISCII/ASCII 8-bit code.

Additional information on ISCII/ASCII 7-bit code and tables showing the relationship of EBCDIC, Hollerith punch code, and ISCII/ASCII 8-bit code is given in Appendix E, "Equivalent ISCII/ASCII, EBCDIC, and Hollerith Codes" on page 133.

## Volume Initialization

Version 3 volumes are initialized with the IEHINITT utility program, 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 80.

## ISO/ANSI/FIPS Version 3 Installation Exits

Five **installation exits** are included in the system for Version 3 volumes:

- WTOR exit for conversion from non-Version 3 to Version 3 labels,
- Volume access,
- File access,
- Label validation, and
- Label validation suppression.

These exits are described in Appendix D, "ISO/ANSI/FIPS Version 3 Installation Exits" on page 131.

## RACHECK (RACF) Installation Exits

The RACHECK preprocessing and postprocessing installation exits can act on Version 3 volume accessibility and label validation. These exits are discussed under "Protecting Data" on page 67 and in Appendix D.

## Volume Protection

Version 3 volumes can be protected by either RACF or the volume access exit. (See "Protecting Data" on page 67 and Appendix D.)

## Data Set Protection

Version 3 data sets can be protected by the file access exit, data set password protection, or a RACF data set profile. (See "Protecting Data" on page 67 and Appendix D.)

## Label Validation

Labels and program specifications for labels and file structure are checked to ensure that a tape format does not violate Version 3 standards. The label validation exit is entered if any of the following requirements are not met:

- Labels must contain only valid ISO/ANSI/FIPS characters. (For a list of the characters, see "Label Definition and Organization" on page 61.)
- Label fields must contain properly aligned data. (For more information, see "Label Definition and Organization.")
- Labels that frame a data set must be symmetrical:
  - DISP=MOD is not allowed for extending an existing data set (however, it may be specified to create a new data set).
  - An open option for EXTEND, OUTINX, or INOUT is not allowed.
  - 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 66, "Creating a Volume Label" on page 80, and *DFP: Customization*.)
- Block size must not exceed 2048 bytes. (See "Format of the Version 3 Data Set Label 2 (HDR2/EOV2/EOF2)" on page 97.)
- 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 volumes.
- Duplicate data set names, including names for generation data sets, are not allowed on the same volume. (See "Processing the HDR1 Label" on page 74.)
- Expiration dates for successive data sets on a volume must not be in ascending sequence. (See "Expiration Date on Existing Label" on page 84.)

### Generation Data Groups

For Version 3, 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 the Version 3 Data Set Label 1 (HDR1/EOV1/EOF1)" on page 91.)

### 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 the Version 3 Data Set Label 2 (HDR2/EOV2/EOF2)" on page 97 and *Data Administration Guide*.)

### Block Padding

A fixed-length logical record (Format-F) containing all ISCII/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, will signal an end-of-block condition.

### Checkpoints

Any ISCII/ASCII tape that is open during a CHKPT macro service will prevent a checkpoint from being taken.

### Compatibility with Non-Version 3 Volumes

For input, volumes with Version 1 labels are accepted, but without Version 3 checking. Volumes with other than Version 1 or Version 3 labels are not accepted for input.

For output, all volumes will be written with Version 3 labels. A volume with Version 1 labels, or any other volume with an 80-character volume label containing the ISCII/ASCII characters VOL1 in the label identifier field, may 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.) The volume label is rewritten and the new header and trailer labels are written in Version 3 format. This converts the volume to Version 3. Note that this conversion requires intervention by the console operator (in response to message IEC704A during open/EOV, or IEC701D with the IEHINITT utility program). Therefore, with careful operational procedures, you can avoid creating an unexpected Version 3 volume.

## Processing Differences between Version 1 and Version 3

Version 3 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 run under Version 3. Therefore, Version 1 users should check their job streams and modify them, if necessary, before installing Version 3 support.

The characteristics for which Version 3 processing differs from Version 1 are:

- Label validation
- Generation data groups

- Spanned records
- Block padding
- Checkpoints
- Compatibility with non-Version 3 volumes

## Processing Version 3 Volumes on Earlier MVS Systems

When a Version 3 volume is mounted on an earlier MVS system, the following restrictions apply:

- Input generation data sets will not be recognized (because the .GxxxxVyy suffix is not part of a Version 3 file identifier).
- Spanned record format will not be supported.
- Data set characteristics recorded in the second header or trailer label of an input volume will not be made known to the application program; instead, they must be specified either in JCL or in the DCB. (This is because the Version 3 MVS system code, IBMZLA, is not recognized by earlier systems.)
- A data set having a header label created by a non-MVS system with a 1 or a 3 in the accessibility field will be treated as an MVS password-protected data set, unless the volume is defined to RACF.
- An uppercase letter from A through Z in the accessibility field of a header or volume label will cause the volume to be rejected.
- Any output produced will not meet the specifications of Version 3 standards.

## Label Definition and Organization

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

- A maximum of 9 user volume labels can appear in the beginning-of-volume group.
- An unlimited number of ISO/ANSI/FIPS user labels may 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/FIPS labels VOL1, HDR2, EOF2, and EOV2 are slightly different from the formats of the corresponding IBM labels.
- ISO/ANSI/FIPS labels HDR2, EOF2, and EOV2 are optional. These labels are required for IBM standard label tapes.
- A maximum of 9 user EOF or EOV labels can appear in the file section label group.

The labels must be recorded in the subset of ISCII/ASCII characters allowed by ISO/ANSI/FIPS standards. These characters are:

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

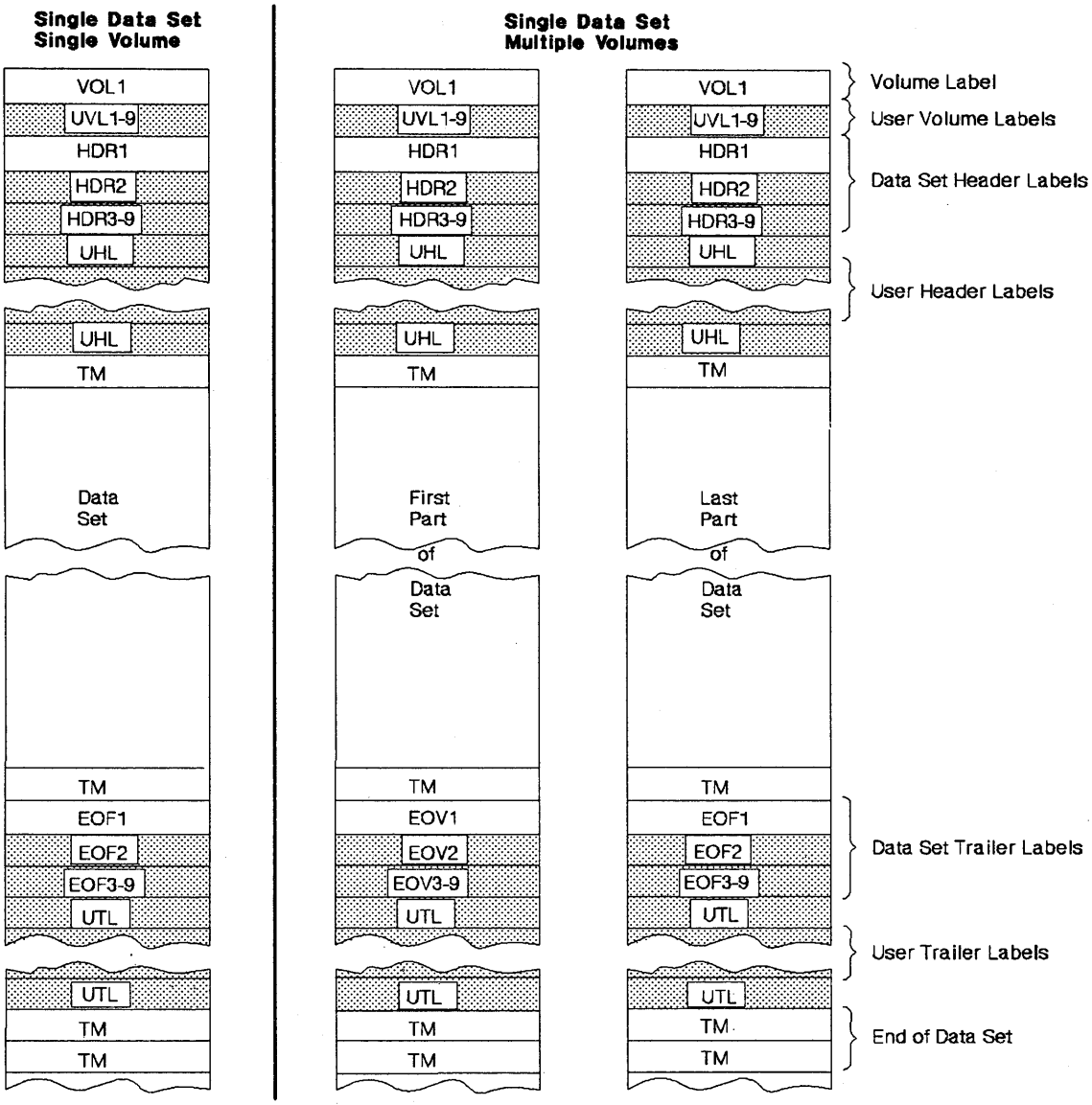
All fields in Version 3 system labels (VOL1, HDR1, HDR2, EOV1, EOV2, EOF1, and EOF2) will be treated as containing meaningful data. This means alphameric fields (except "reserved for operating system" fields) will be left-justified, with unused positions filled with space characters; "reserved for future standardization" fields will be filled with space characters; numeric fields will be right-justified, with unused positions filled with zeros. Date fields will have a leading space character.

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

| **Label Identifier** | **Label Definition** |
|---|---|
| VOL1 | Volume label |
| UVL1-UVL9 | User volume labels (optional; *not* 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) |
| 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) |
| UHLa[1] | User header labels (optional; unlimited number permitted) |
| UTLa[1] | User trailer labels (optional; unlimited number permitted) |

[1] The fourth character of the user header and trailer labels may be any valid ISO/ANSI/FIPS character as defined above.

Figure 11 on page 63 and Figure 12 on page 64 show the position of the labels with various tape volume organizations. User labels (UHL, UTL) are optional.

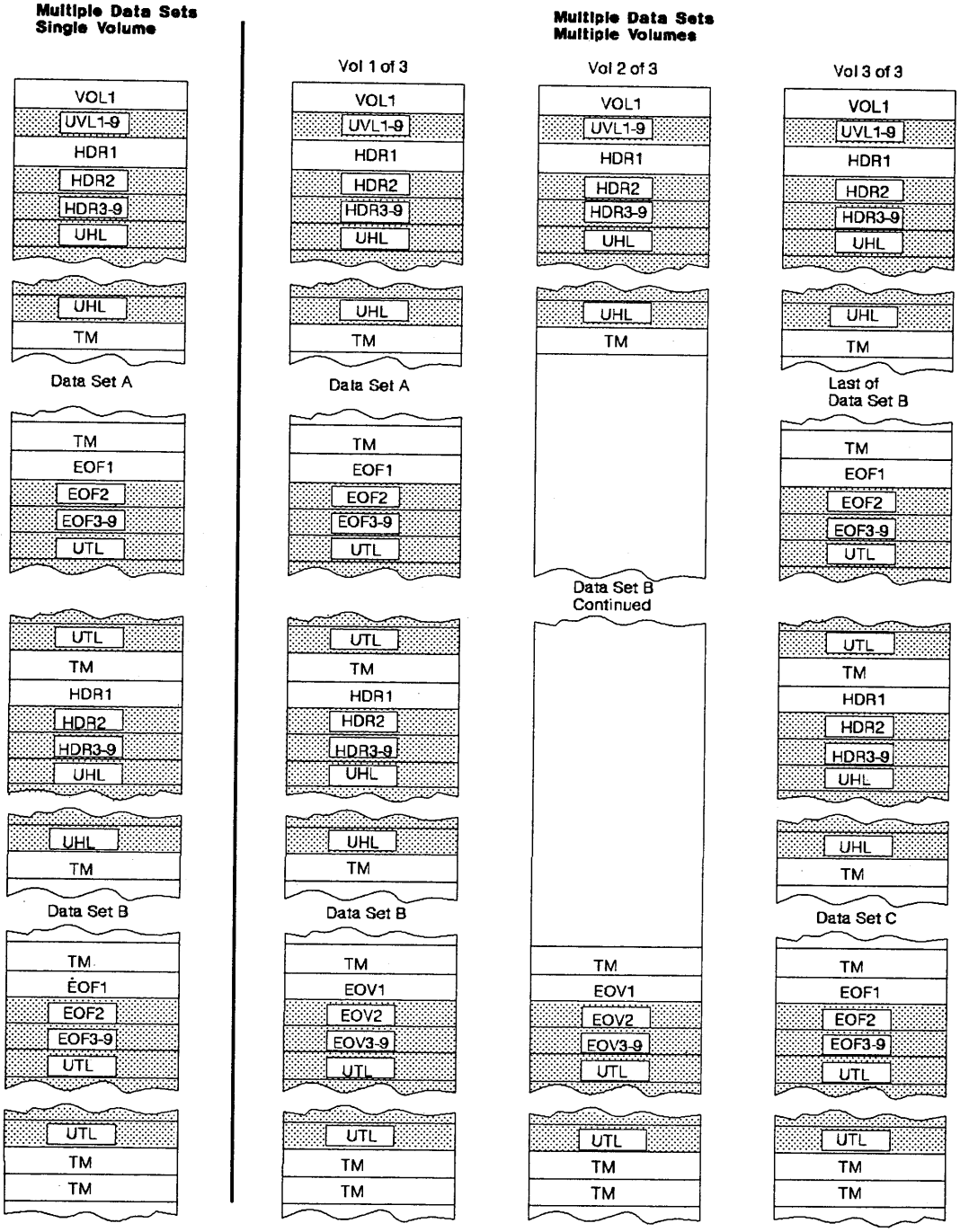



**Single Data Set/Single Volume**: The volume label is followed by the data set header labels and optional user header labels. The data set is preceded and followed by a tapemark. The data set trailer labels are identified as EOF and followed by optional user trailer labels. Two tapemarks follow the 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.

**Note**: Shading indicates optional labels for ISO/ANSI labeled tapes.

Figure 11. Volume Organization with ISO/ANSI/FIPS 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 data set trailer labels are identified as EOV instead of EOF, and the trailer label group is followed by two tapemarks. There is a separate volume label for each tape.

**Note**: Shading indicates optional labels for ISO/ANSI labeled tapes.

Figure 12. Volume Organization with ISO/ANSI/FIPS Labels (Multiple Data Sets)

## Volume Label

The volume label (VOL1) is the first record on an ISO/ANSI/FIPS labeled tape. It is at least 80 characters long. Although an ISO/ANSI/FIPS 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 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 data management to verify that the correct volume is mounted. It also protects the volume from unauthorized use.

Volume labels are created by the IEHINITT utility program, or by input from the operator during open 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), or
- The first block of a volume cannot be read (except a RACF-protected volume will be 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 the IEHINITT utility program, see *Utilities*.

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/FIPS 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 at least 80 characters long, and any characters after position 80 are not used for label checking and verification.

A HDR2 label is optional for ISO/ANSI/FIPS labeled tapes. If 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.

User header labels (UHLa) are optional, and there is no limit to the number that can be associated with a data set. They must contain only valid ISO/ANSI/FIPS characters, but need not be sequentially numbered or lettered.

Labels identified as HDR3-HDR9 are allowed but are 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.

## Data Set Trailer Labels

The data set trailer label 1 (EOV1, EOF1) is required for ISO/ANSI/FIPS 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).

Eighty-character Version 3 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 at least 80 characters long; any characters after position 80 are not used for checking or verification.

Although they are optional for ISO/ANSI/FIPS 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, the "reserved for operating system" fields are not processed.

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 valid ISO/ANSI/FIPS characters, but need not be sequentially numbered or lettered.

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

## Tape Marks

The tape mark requirements for ISO/ANSI/FIPS interchange tapes are:

- Each header label group must be followed by a tape mark that indicates the beginning of the data to be processed.
- Each data set must be followed by a tape mark 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 tape marks. `Otherwise, one tape mark is required.

When creating a data set with Version 3 labels, the system routines write the necessary tape marks.

## 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 volumes can be protected by means of one of the following:

- RACF.
- IBM-supplied Version 3 installation exits described in Appendix D, "ISO/ANSI/FIPS Version 3 Installation Exits" on page 131. (These can be replaced by installation-written exit routines.)
- Data set password protection.
- A combination of the installation exits and password protection.

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

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 (for information about the program properties table, see *System Modifications*).

If a volume is RACF protected, ALTER access authority is required to create or destroy the VOL1 label, UPDATE access authority is required to open the volume for output (this includes reading and/or writing on the volume), and READ access authority is required to open the volume for input.

If the tape volume is not defined to RACF, or if the tape volume is defined and the user is UPDATE authorized and PROTECT=YES has not been specified, processing continues. If the tape volume is defined and either the user is UPDATE authorized and PROTECT=YES has been specified, or the user is not UPDATE authorized, the volume will be rejected if a nonspecific request is made and the program will be abnormally terminated if a specific request is made. The operator will be requested to remove the write-enable (file protect) ring from a RACF-protected tape when a user with only READ authority attempts to process a tape for input. For an overview of RACF protection for tape volumes, see *RACF General Information*.

Data set password protection is described in *System—Data Administration*.

### Volume Accessibility

*Version 3 Volumes:* If the volume is RACF protected, and 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 an uppercase letter from A through Z, the RACHECK parameter list is initialized to address the Version 3 exit parameter list (IECIEPRM) and the accessibility code. Otherwise, the RACHECK installation exits are passed zeroed pointers for the Version 3 parameters. If RACF accepts the volume but the Version 3 parameters to the exit were 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 an uppercase letter from A through Z, 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 IEHINITT utility, or by the operator during open (see "Volume Label" on page 65).

*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, the RACHECK installation exits are entered to accept or reject the volume (by a return code from RACHECK). If the field contains a space character, but the volume is not defined to RACF or volume security processing has been suppressed, access to the volume is allowed (however, as will be discussed below, each data set on the volume may be individually protected). If the field contains anything other than a space character, the volume is rejected.

The VOL1 accessibility code can be changed when the VOL1 label is changed by the IEHINITT utility, or by the operator during open (see "Volume Label" on page 65).

## Data Set Accessibility

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

*Version 3 Volumes:* A 1 or a 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 an uppercase letter A through Z 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 if the first character of the JCL ACCODE keyword is A through Z). Any other character causes the volume to be rejected.

If the accessibility field of a data set being opened contains an A through Z 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 may 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. You should 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 A through Z 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 through Z can be monitored by an installation-written file access exit. Password codes override ACCODE values if they are simultaneously specified.

*Version 1 Volumes:* A 1 or a 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

For input processing of a password-protected data set, data management asks the operator or TSO user to enter the correct 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 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.

*System—Data Administration* describes the protection feature and discusses how to create and maintain the password data set.

## Output Processing for Password-Protected Data Sets

For output processing of a password-protected data set, data management compares the data set name shown in the HDR1 label with the data set name specified in the DD statement. 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 will be requested to demount the tape and mount a new scratch tape. If the names are the same, data management 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, you must 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 has been defined to RACF.
- 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 and that the protection-mode indicator allows writing to the data set.

*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. This can be done by using either the IEHINITT utility or a user program to relabel the volume.

## Overview of ISO/ANSI/FIPS Label Processing

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

- Translating input labels from ISCII/ASCII to EBCDIC and translating output labels from EBCDIC to ISCII/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 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 standards
- Creating and writing new labels on output tapes

These processing functions are summarized in Figure 13 on page 71. The table shows the specific labels that are processed for each function and which routines perform the functions.

Although the default of each of the IBM-supplied installation exits is to reject the volume, the exits may be modified to do additional label processing (see Appendix D, "ISO/ANSI/FIPS Version 3 Installation Exits" on page 131).

All volumes will be written with Version 3 labels. For information about using tapes with other than Version 3 labels, see "Compatibility with Non-Version 3 Volumes" on page 60.

| Processing | Volume Label | | Header Labels [1] | | | | Trailer Labels [1] | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | VOL1 | User Volume Labels | HDR1 | HDR2 | HDR3–9 | UHL | EOF1 or EOV1 | EOF2 or EOV2 | EOF3–9 or EOV3–9 | UTL |
| **First or only volume: 2** | | | | | | | | | | |
| Checks labels on input tape. | Open | ignored[3] | Open | Open | ignored[3] | Open | EOV | bypassed | ignored[3] | EOV |
| Checks existing labels on output tape before overwriting. | Open | ignored[4] | Open | not read | not read | not read | not read | not read | not read | Open |
| Writes new labels on output tape. | Open, or user[5] | not written | Open | Open | not written | Open | Close or EOV | Close or EOV | not written | Close or EOV |
| **Second or subsequent volumes: 3** | | | | | | | | | | |
| Checks labels on input tape. | EOV | ignored[3] | EOV | bypassed | ignored[3] | EOV | EOV | bypassed | ignored[3] | EOV |
| Checks existing labels on output tape before overwriting. | EOV | 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 | EOV | EOV | not written | EOV | Close or EOV | Close or EOV | not written | Close or EOV |

**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. (Data sets with like characteristics can be processed correctly using the same data control block (DCB), input/output block (IOB), and channel program. Any exception in processing makes the data sets unlike.)

[3] Label sequence is checked but contents are ignored.

[4] Operator must give Open/EOV permission to overwrite ULV labels.

[5] User creates the label with the IEHINITT utility program or a user program.

[6] Includes the first volume of concatenated data sets with like characteristics.

Figure 13. ISO/ANSI/FIPS Standard Label Processing by Data Management Routines

## 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 has specified AUL in JCL.

The tape device is checked for a density compatible with the density specified by the user, and the system is checked to ensure that the ASCII option was specified in the CTRLPROG macro at system generation time.

Before verifying the volume, if the system-wide RACF tape protection option has been specified and the DD statement has specified PROTECT=YES without previously opening the DD statement for output processing, the program will be abnormally terminated.

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 ISCII/ASCII identifier VOL1 in the first 4 bytes. The volume label can be either Version 1 or Version 3. If the record is over 80 bytes long, the excess characters are ignored.

After the VOL1 label is read, a work area is allocated to contain ISO/ANSI/FIPS data, such as the exit parameter list. Information from the VOL1 label is saved in the UCB class extension.

Data management 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), 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 labels is not RACF protected and the VOL1 accessibility field contains an uppercase letter from A through Z, the volume access exit is entered. For more details of RACF and accessibility checking, see "Protecting Data" on page 67.

For a tape with Version 3 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 validation has been suppressed). In addition, unless validation has been suppressed, the VOL1 label is checked for conditions not allowed by Version 3 standards. If an invalid condition is found, the label validation exit is entered.

## Volume Serial Number

Data management 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. 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 or RDBACK.

If the volume serial number is correct, the volume is considered to be mounted and verified. If the serial number is not correct, data management rejects the tape and issues another mount message.

## Positioning the Volume to the Data Set

When the volume is mounted and verified, data management 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 reel, 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 reel of tape. 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, your task will be abnormally terminated or, if the volume ends with EOV 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, data management assumes that the data set is the first in sequence on the volume.

To position the tape, data management 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 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 EOV 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. 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 tape mark.

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 EOV 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 will be set to X'0000' if:
   - The data set was never opened or,
   - CLOSE (,REWIND) was specified or,
   - CLOSE (,REREAD) and LABEL=1 was specified or,
   - 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 abends 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) tape mark during open. This is usually done by reading data but suppressing data transmission to storage until a tape mark 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) will be given control to allow recovery. For more information about the "data management abend installation" exit, see *DFP: Customization*.

Only one data set on a tape volume may be open at any given time. An attempt to begin processing of 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, data management checks the label identification. If the identifier HDR1 is not found, processing is abnormally terminated.

## Processing the HDR1 Label

To ensure that the correct data set is being opened, data management compares the file identifier field in the HDR1 label of the requested data set to the data set name specified in the DD statement. If the comparison shows an incorrect data set name, processing is abnormally terminated.

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 data management; however, for Version 3 HDR1 labels, they are not included in the file identifier. For generation data sets with Version 3 labels, the generation and version numbers do not participate in password protection.

For tapes with Version 3 labels, during positioning, the file identifier in each HDR1 label encountered is also checked to ensure 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 will occur 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 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, the label validation exit is entered. Note, however, that, 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 will be entered with a "duplicate name error" condition. If the exit accepts the error, an abend may result when the target data set does not match the requested data set name.

For tapes with Version 3 labels, the accessibility field in the HDR1 label is also checked if the volume is not RACF protected. If it contains an uppercase letter from A through Z, the file access exit is entered. (For more details about accessibility, see "Protecting Data" on page 67.)

The expiration date shown in the HDR1 label is not verified for input data sets, except to check the field in a Version 3 label for valid format.

The block count shown in the HDR1 label is always an ISCII/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 end of data set or end of volume.

### Processing the HDR2 Label

Because the HDR2 label is optional with ISCII/ASCII interchange tapes, and because 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, the HDR2 label "Reserved for Operating System" field contains the same data as an IBM standard HDR2 label, and is processed accordingly (see "Data Set Characteristics" on page 24). 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.

Unless user header labels are to be processed, data management reads forward until a tape mark 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.

### 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 DCB exit list (EXLST). (The DCB exit list (EXLST) is described in *DFP: Customization*.) If you omit one of these parameters, data management positions the tape past the tape mark immediately after processing the HDR2 label, or the HDR1 label if the HDR2 label is not present.

### Read Backward

For the read backward (RDBACK) processing method, data management 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

Data management's EOV routine handles both end-of-data-set and end-of-volume conditions on input. These conditions occur when:

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

After encountering a tape mark, data management checks the first 4 bytes of the first trailer label for the identifier EOV1 or EOF1. If neither identifier is found, processing is abnormally terminated.

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 read backward operation (RDBACK), data management ignores the second trailer label (EOV2 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 standards.

When the FEOV macro instruction 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 user trailer labels (UTL) are present on input, data management 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

To verify that all records on the input data set on the current volume have been read, data management 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, data management 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 0B (hexadecimal). 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:

| Return Code | Meaning |
|---|---|
| 0 | Abnormally terminate with completion code 237 |
| 4 | Continue processing |

If you do not provide the appropriate user exit entry in the DCB exit list, a block count discrepancy causes processing to abnormally terminate with a completion code of 237.

When the FEOV macro instruction is executed, the block count is not verified.

## 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. 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. Data management 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 volume sequence number in the VOLUME parameter of the DD statement.

For input, the label identifier of the trailer labels determines whether data management continues processing the data set. When data management finds an EOV label, it performs volume switching. When data management finds an EOF label, it passes control to the user's end-of-data routine, with one exception: If the DD statement specifies OPTCD=B, data management performs volume switching.

To determine whether additional volumes are required, data management maintains a volume sequence number in the data extent block (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-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), data management 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 premounting 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 don't want message retention, turn off the message retention attribute. You can accomplish this by using an exit such as IEECVXIT, IEAVMXIT, or other user exit to alter the message descriptor code to type 3. For additional information on these user exits, see *User Exits* and *System Modifications*.

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 DOM (delete operator message) to be initiated against the IEC501E message and will issue 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 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. Volume accessibility and data set accessibility are also checked as if it were the first volume of the group.

If a new concatenated data set is not processed and the open is for OUTIN, it will be further verified that, if the new volume is RACF defined, it is defined as part of the same volume set profile as the previous volume.

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 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 instruction is executed, processing of trailer labels is not performed.

## Creating a Volume Label

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

The IBM-supplied utility program IEHINITT writes the following labels in ISCII/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 will contain zeros, with a leading space in the date fields.
3. A tape mark.

A tape initialized with these labels should not be confused with an ISO/ANSI/FIPS tape, which requires at least one data set (the data set can be empty).

Version 3 standards require label symmetry around an empty data set when a volume is initialized. The labels written by IEHINITT are accepted by data management routines, which, in turn, produce symmetrical labels; the HDR1 label is updated with system data, the single tape mark is overwritten with a HDR2 label containing data set characteristics, and a tape mark is written to complete the beginning-of-volume and beginning-of-file label groups. Whether or not any data is written in the data set, a set of trailer labels is written when the data set is closed.

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. Detailed procedures for using the program are described in *Utilities*.

Methods other than the IEHINITT utility program 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 (see Chapter 6, "Volume Label Verification and Volume Label Editor Routines" on page 115) with one that writes VOL1 labels. Some data or a tape mark should already exist on the tape; otherwise, the tape control unit may read through the entire reel of blank tape looking for some data.

Except for the IBM 3480 Tape Subsystem, the VOL1 label is rewritten by the open or EOV routines if **all** the following conditions are met:

- A density conflict has occurred.
- OUTPUT or OUTIN is specified in the OPEN macro instruction.
- 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.
- The volume does not have UVL labels. If it does, the operator must give permission to rewrite the VOL1 label because the UVL labels will be overwritten.

On an IBM 3480 Magnetic Tape Subsystem the open and EOV modules bypass rewriting a VOL1 label.

If you request output to the first data set on an ISO/ANSI/FIPS output volume (AL, AUL) and the tape that you are allocated is recorded in the wrong density and cannot be read, the open or EOV routine rewrites the VOL1 label in the density you specify.

This allows you to make nonspecific requests for output tapes (that is, you need not specify a volume serial number in your DD statement) and allows the operator to mount any available scratch tape to satisfy 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 request output to other than the first data set on the volume and a density conflict occurs, all further processing is done in the density of the existing VOL1 label (unless the IBM-supplied volume label editor routines are replaced with installation-written routines that do otherwise; note, however, that, if the densities are mixed, the tape will not meet the specifications of Version 3 standards).

If you make an output volume request for an ISO/ANSI/FIPS labeled (AL, AUL) tape and the mounted volume is an NL, NSL, SL, or SUL labeled tape, the open or EOV routine will create a VOL1 label only after checking authorization for access, checking the expiration date, and sending a message to the console

operator requesting serial number and owner information (see Figure 14 on page 89, Fields 3 and 7).

## Opening an Output Data Set

When a data set is opened for output, processing is similar to that of opening for input. The exceptions are:

- Only Version 3 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, and the DD statement has not previously been opened for output processing, OPEN ensures the PROTECT=YES specification is valid; both the volume sequence number and the file sequence number must be set to one 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 will not attempt validity checking (of the PROTECT=YES specification) and definition of the volume to RACF.
- The volume label editor is entered for label conflicts. The volume label editor is also entered for version conflicts (the label is not Version 3) if output is to the first data set; if output is to any other than the first data set or if the first data set is allowed to be extended, a version compatibility conflict will cause 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 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 67.
- Symmetry violations during output to a Version 3 volume will 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 will be 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) 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 reverification of the volume, or
- The operator rejects a rewrite of the label by a response to message "IEC704A L" or "IEC704 L UVL."

New header labels are written after the volume label is rewritten.

**CAUTION:**
**Doing multiple opens and closes without writing any user data in the area of the end-of-tape reflective marker may result in the header and trailer labels being written past the marker. Access methods detects the markers, but because the creation of empty data sets does not involve access methods, the end-of-tape marker will never be detected. This may cause the tape to run off the end of the reel.**

## Volume Serial Number

You need not specify a volume serial number for output tapes. If none is specified, the mount message directs the operator to mount a scratch tape. Data management gets the volume serial number from the VOL1 label and records it in the JFCB and UCB.

If you choose to specify the volume serial number, data management compares it with the volume serial number shown in the VOL1 label. If the number is correct, data management resets a mount switch in the UCB to indicate that volume mounting is verified (the switch is initially set when the mount message is issued to the operator). If the volume serial number is incorrect, data management 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, data management rejects the tape and issues another mount message.

## Positioning the Volume to the Data Set

When the volume is mounted and verified, data management 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 EOV 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 will not include the explicit sequence number; instead, it will indicate "254+."
- If you do not specify a sequence number, or if you specify 0, data management assumes that the data set is to be written as the first one on the volume.

To position the tape, data management 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. If you attempt to open another data set on the same volume, processing is abnormally terminated. This restriction includes system output (SYSOUT) tapes.

When the tape is positioned to receive the new data set, data management expects to find either an existing HDR1 label or a tape mark. If neither is present, data management 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 EOV labels, another data set cannot be written to follow it.)

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

- When a tape mark immediately follows the VOL1 label. This may occur when the tape is initialized by means other than the IEHINITT utility program (IEHINITT writes a dummy HDR1 label following the VOL1 label). The tape mark 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, data management encounters the second tape mark following the existing EOF trailer label group. The tape mark 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 71.

If a volume is not RACF protected, the accessibility code in the existing HDR1 label of a Version 3 volume is checked before it is overwritten with a new HDR1 label.

## Expiration Date on Existing Label

For a volume with multiple data sets, data management checks the expiration date of the data set that immediately precedes the output data set. If the previous data set's expiration date is later than the expiration date of the output data set, the label validation exit is entered, unless label validation has been suppressed.

If the previous data set's expiration date is equal to or greater than that of the output data set, 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, the operator is asked to confirm use of the tape or to mount another tape.

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 data management for output, data management 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 Chapter 1, "Introduction to Tape Processing" on page 1.

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

### Writing User Header Labels

For data management to write user header labels (UHL), you must code AUL on the DD statement and specify the address of an output user header label routine in the DCB exit list (EXLST). The DCB exit list (EXLST) is described in *DFP: Customization*.

Unless label validation has been suppressed, the user header labels will be validated to ensure they contain valid ISO/ANSI/FIPS characters. Violations will 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, the operator will be requested to demount the tape and mount a scratch tape, even if you request 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 will be abnormally terminated.

## End-of-Volume on Output

The data management EOV routine automatically switches volumes when an end-of-volume condition occurs, that is, when a reflective strip is encountered or a FEOV macro instruction is executed. This volume switching includes:

- Writing trailer labels on the current volume
- Checking existing volume and header labels on the new volume
- In the case of a density conflict, writing a volume label on the new volume
- Writing header labels on the new volume

In the case of a density conflict when the volume label is checked, a new label (Version 3) is written in the density specified by the user. If any user volume labels (UVLs) exist, they will be 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 being used, the EOV routine 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 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 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 don't want message retention, turn off the message retention attribute. You can accomplish this by using an exit such as IEECVXIT, IEAVMXIT, or other user exit to alter the message descriptor code to type 3. For additional information on these user exits, see *User Exits* and *System Modifications*.

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 DOM (delete operator message) to be initiated against the IEC501E message and will issue the IEC501A message.

## Writing Data Set Trailer Labels

Trailer labels are always written at an end-of-volume condition on output tapes and are identified as EOV1 and EOV2 (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, two tape marks are written following the data set trailer labels. If user trailer labels are to be written, the tape marks 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 DCB exit list, data management can write as many user trailer labels as desired.

The contents of user trailer labels are not checked during EOV processing, but, if they contain any invalid ISO/ANSI/FIPS characters, the condition will be 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.

When creating 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 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 will be 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 labels, the volume access exit is entered if the new volume is not RACF protected and the accessibility field in the VOL1 label contains an uppercase letter from A through Z.

### Special End-of-Volume Conditions

When a reflective strip causes an end-of-volume condition during the writing of data, the EOV routine writes the trailer labels as described above. If the reflective strip is encountered while 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, even though it crosses the reflective strip.

If you add another data set to a tape (multiple data set organization) on which the last existing trailer label group crossed the reflective strip, or on which the new header label group crosses the reflective strip, data management:

- 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 or OUTIN) or when no output is written before closing (for OUTPUT or OUTIN), 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 tape marks following the trailer labels. If user labels are to be written, the tape marks follow the user trailer labels. If another data set is added to the tape (multiple data set organization), its HDR1 label overlays the second tape mark.

### 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 DCB exit list, the close routine can write as many user trailer labels (UTL) as desired.

The contents of user trailer labels are not checked during close processing, but, if they contain any invalid ISO/ANSI/FIPS characters, the condition will be detected during a subsequent open for RDBACK.

## IBM 3480 Tape Processing

To improve performance when processing labels of ISO/ANSI/FIPS labeled volumes on an IBM 3480 Tape Subsystem, the open and EOV routines attempt to avoid changing the direction of tape movement.

## Checkpoint/Restart Not Allowed

Any ISCII/ASCII tape that is open during a CHKPT macro service will prevent a checkpoint from being taken.

## Format of the Version 3 Volume Label (VOL1)

Figure 14 on page 89 shows the format of the Version 3 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, as are differences between the Version 3 volume label and the IBM volume label.

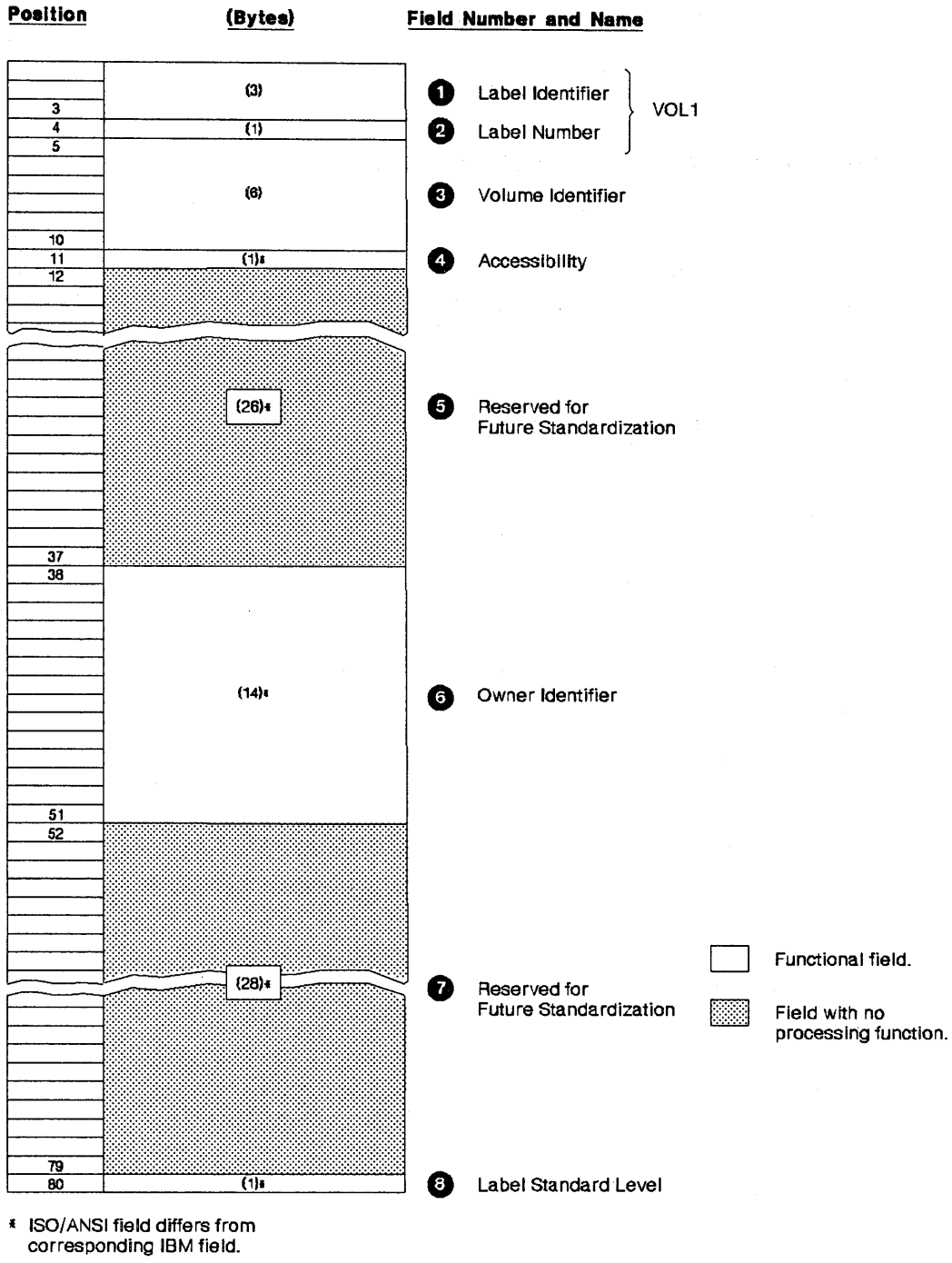



Figure 14. Format of the Version 3 Volume Label

**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 IEHINITT utility program or a user's program.

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

- If you request a Version 3 output volume (AL, AUL), and an NL, NSL, or SL labeled volume is mounted to satisfy your request
- If you request a Version 3 output volume (AL, AUL), and a volume that can be overwritten and that is recorded in the wrong density is mounted to satisfy your request (only if output is for the first data set on the volume)

**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 the Version 3 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, that is assigned through JCL or IEHINITT to the volume when it enters the system, or that is assigned by the operator when open or EOV routines label the volume. This code may also appear on the external surface of the volume for visual identification. The code is normally numeric (000001 to 999999), but can be from 1 to 6 characters long. If the code is less than 6 characters long, it must be left-justified, and will be padded with blanks.

  All national characters and some special characters in this field will be rejected during open as invalid ISO/ANSI/FIPS characters. For a list of the valid ISO/ANSI/FIPS characters, see "Label Definition and Organization" on page 61.

- Processing: The user-specified volume serial number is obtained from the JFCB and recorded in the UCB. Then the number in the UCB is compared 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: An uppercase letter from A through Z indicates that the RACHECK installation exits will be entered and will 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 will cause the volume to be rejected if the volume is not defined to RACF.
- Processing: If this field contains any character other than a space or an uppercase letter from A through Z, 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** (26 bytes)

- Contents: Reserved for possible future use; should be recorded as blanks.
- Processing: Not used or verified, except to check for all blanks. The IEHINITT utility program writes blanks in this field. (Blanks are translated to ISCII/ASCII space characters on output.)

**6—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 ISO/ANSI/FIPS characters. The IEHINITT utility program writes the text specified by the user, and the open and EOV routines 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 ISCII/ASCII space characters on output.)
- Difference from IBM Field: The corresponding field on an IBM standard label is 10 bytes long.

**7—Reserved for Future Standardization** (28 bytes)

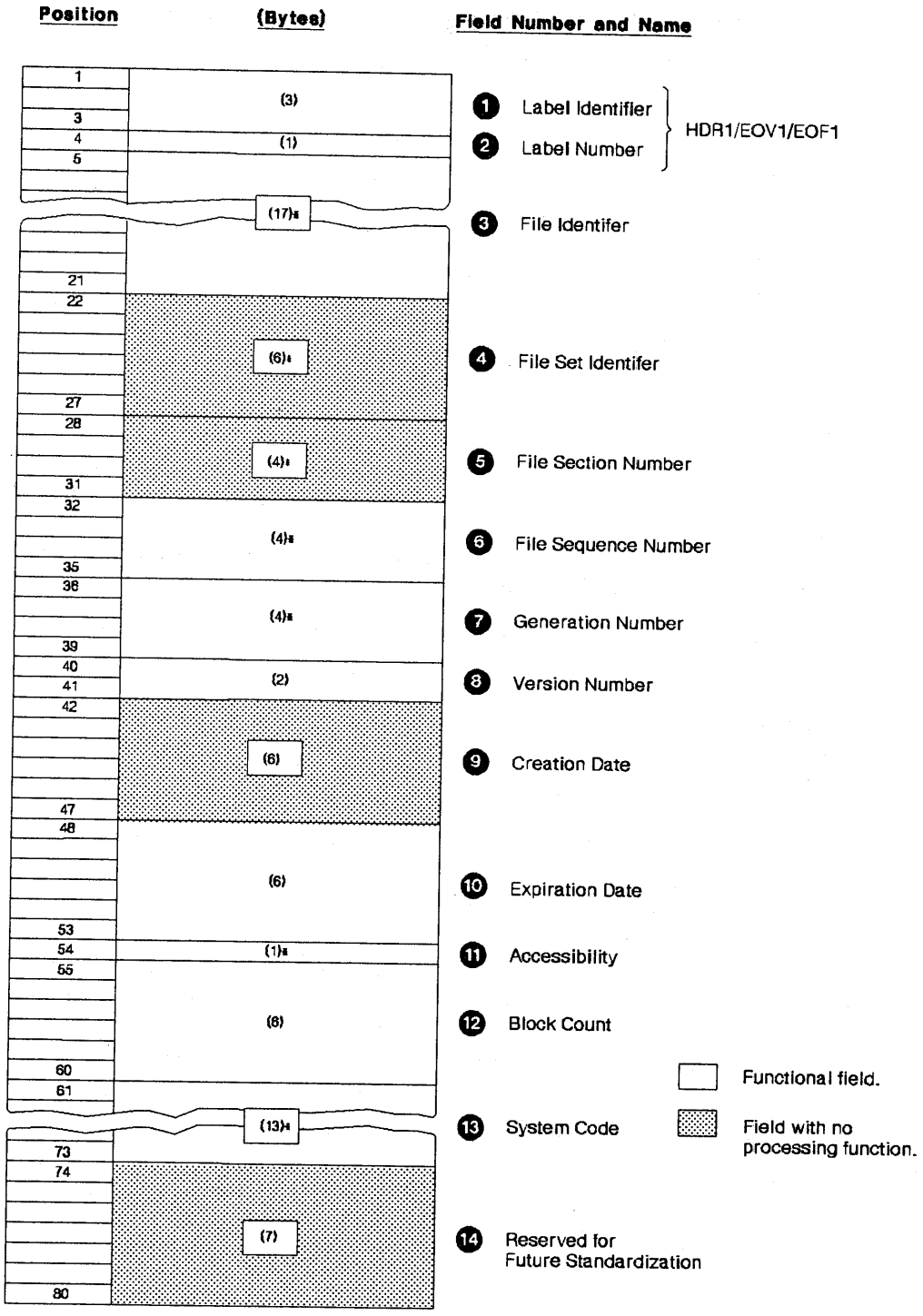
- Contents: Reserved for possible future use; should be recorded as blanks.
- Processing: Not used or verified, except to check for all blanks. The IEHINITT utility program writes blanks in this field. (Blanks are translated to ISCII/ASCII space characters on output.)

**8—Label Standard Level** (1 byte)

- Contents: A 3 signifies that the tape is formatted according to Version 3 interchange standards.
- Processing: The operating system will always place a 3 in this field on output. Version 1 tapes contain a 1 in this field and are accepted for input processing. Any character other than a 1 or a 3 in this field is rejected by the system.
- Difference from IBM Field: This field is blank in IBM standard labels.

## Format of the Version 3 Data Set Label 1 (HDR1/EOV1/EOF1)

Figure 15 on page 92 shows the format of HDR1, EOV1, and EOF1. 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, as are differences between the Version 3 labels and the IBM labels.




* ISO/ANSI field differs from corresponding IBM field.

Figure 15. Format of Version 3 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: Data management checks this field to verify that the record is an ISO/ANSI/FIPS data set label.

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

  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, data management performs the switching. If no volumes are available, data management passes control to the user's end-of-data routine.

  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—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 ISCII/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. *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. It 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 74 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, the data set name is not checked, except for valid ISO/ANSI/FIPS characters.

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 assigned in the JCL statements, all national characters and some special characters will be rejected during open as invalid ISO/ANSI/FIPS characters. See "Label Definition and Organization" on page 61 for a list of the valid ISO/ANSI/FIPS characters. If the code is fewer than 6 characters long, it must be left-justified.

- Processing: Not used or verified, except to check for valid ISO/ANSI/FIPS characters. 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: Not used or verified, except to check for valid ISO/ANSI/FIPS characters. When creating labels, the open routine writes 0001 in this field; the EOV and close routines obtain the current volume sequence number from the DEB.

- 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, 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 EOV routine obtains this number from the logical data set sequence number in the UCB.

**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 invalid, and will cause the label validation exit to be entered unless validation has been suppressed.

  When creating labels, data management 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 ISCII/ASCII zeros.
- Processing: Data management 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: Year and day of the year when the data set was created. The date is shown in the format cyyddd, where:

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

- Processing: Not used or verified, except to check for proper format. When data management creates labels, the date is obtained from the JFCB. This is the date the job was initiated for execution, and not necessarily the date the label was created.

**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 cyyddd, where:

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

- 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 shown in the communications vector table (CVT). If the date in the label is later than the current date, the operator receives a message and is given the option of using the tape or mounting another. If other data sets are on the same volume, data management 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, the label validation exit is entered (unless label validation has been suppressed). Any data sets following on the volume are treated as if they had expired on the same day as the output data set.

  When creating labels, data management obtains the expiration date from the JFCB. If you did not specify a retention period or expiration date, the expiration date is recorded as zeros and the data set is considered expired.

**11—Accessibility** (1 byte)

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

| | |
|---|---|
| Uppercase A-Z | If the volume is not RACF protected, the file access exit will be 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.) This 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, data management 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, data management verifies the password furnished by the operator or TSO terminal user and sets a security indicator in the JFCB. If an uppercase letter from A through Z is specified and the volume is not defined to RACF, the file access exit will be entered (the IBM-supplied exit will reject the volume). If a character other than an ISCII/ASCII space, an uppercase letter from A through Z, a 1, or a 3 is encountered, the DCB will not be opened and no further processing will take place.

  For output, data management inspects this field in the existing HDR1 label. If a 1 or a 3 is specified, with the system code "IBMZLA", the existing data set cannot be overwritten until data management 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 is requested to demount 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 an uppercase letter from A through Z is specified and the volume is not defined to RACF, the file access exit will be entered (the IBM-supplied exit will reject the volume).

  When data management 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, which will be maintained internally in an SWB control block. 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 zero (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 will continue or abnormally terminate. If the appropriate user exit entry is not provided, a block count discrepancy causes processing to abnormally terminate.

  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 data management 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 during close and EOV label creation.

  If the data set was created with a 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 OS360 or OS370 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," an d 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, data management writes blanks in this field. (Blanks are translated to ISCII/ASCII space characters on output.)

## Format of the Version 3 Data Set Label 2 (HDR2/EOV2/EOF2)

Figure 16 on page 98 shows the format of HDR2, EOV2, and EOF2. 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, as are differences between Version 3 labels and IBM labels.

If the labels are produced by the operating system, they are treated like IBM header 2 and trailer 2 labels. If the labels are produced by another system, the "Reserved for Operating System" and "Record Format" fields will not be used.

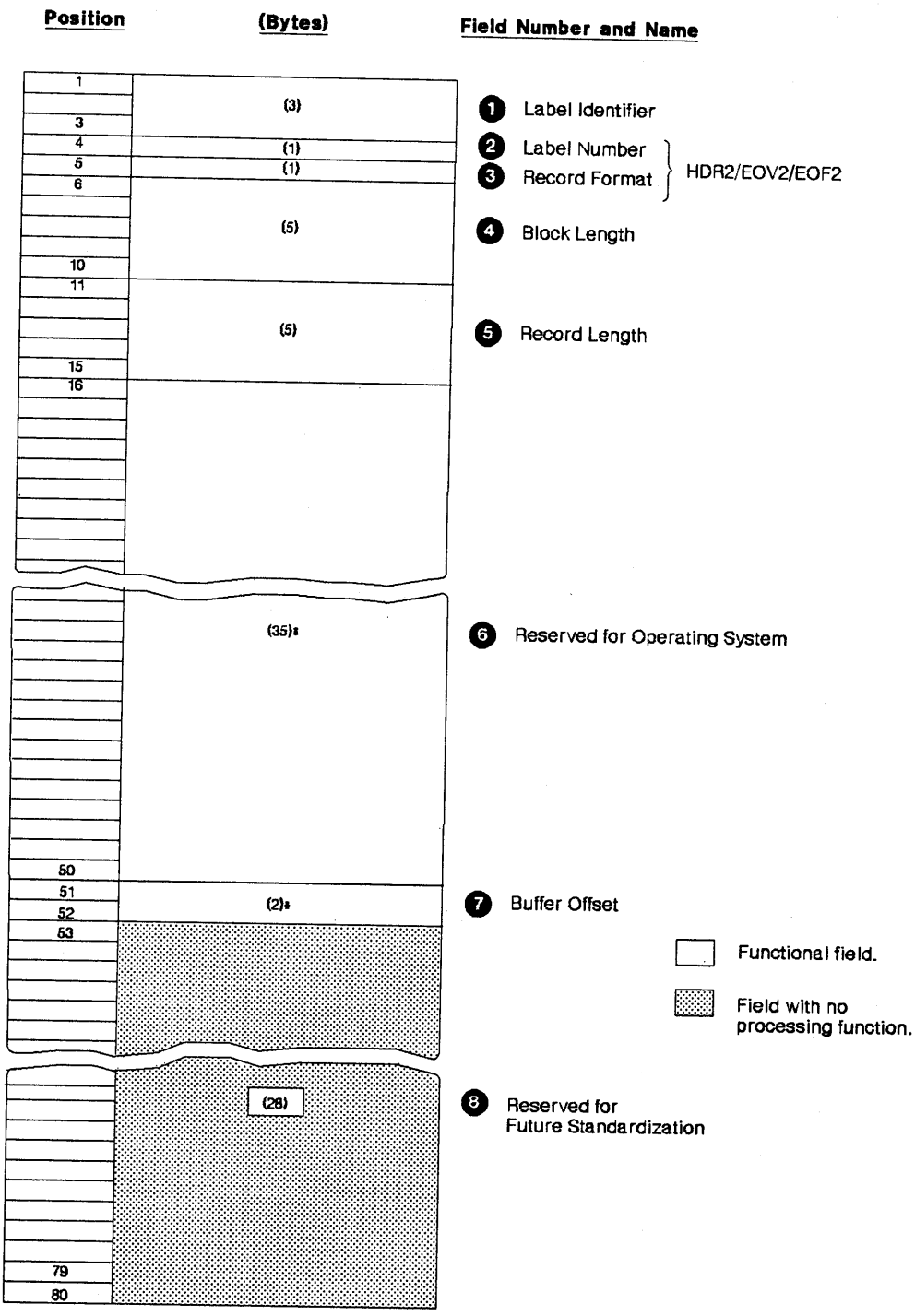



Figure 16. Format of Version 3 Header 2 and Trailer 2 Labels

**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: Data management checks this field to verify that the record is an ISO/ANSI/FIPS data set label.

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

  If the DD statement specifies OPTCD=B for an input data set, data management 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, data management performs the switching. If no volumes are available, data management passes control to the user's end-of-data routine.

  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 tapes, the label validation exit is entered.

  For detailed information about record formats, see *Data Administration Guide*.
- Processing: For input, the record format is obtained from this label and recorded 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, a reverse merge follows the forward merge described above. 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 the introduction.

**4—Block Length** (5 bytes)

- Contents: A number from 18 to 2048 that indicates the block length (including buffer offset and padding) in bytes.

  **Note:** The 18-byte to 2048-byte limit on block length is an ISCII/ASCII standard. Larger blocks (up to 9999 bytes) 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.

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

  - Format F Block length.
  - Format D Maximum block length (including the 4-byte length indicator in the records and the optional block prefix).
  - Format S Maximum block length (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 (including the 4-byte length indicator in the records and the optional block prefix).
  - Format S Maximum record length (excluding all the 5-byte segment control words that describe the record). If the record is larger than 99999, this field is zero.

- 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 following 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 10).
    - Processing: If DCB=DEN was not specified in JCL, this data is merged in the JFCB for input. When data management 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 EOV routine writes a 1. The close routine determines which code to write by comparing the 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.
  - 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 9th byte is a slash (/), and the last 8 bytes contain the name of the job step.
    - Processing: Not used or verified. When data management creates labels, the names of the job and job step are obtained from the TIOT.
  - Tape Recording Technique (2 bytes)
    - Contents: Recorded as blanks for 9-track tape; 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 ISO/ANSI/FIPS 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 ASCII 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, data management 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.

**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 four 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 block descriptor words, see *Data Administration Guide*.)
- 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. (Blanks are translated to ISCII/ASCII space characters on output.)
- Processing: Not used or verified. When creating labels, data management writes blanks in this field.

## Format of Version 3 User Header and Trailer Labels (UHL/UTL)

Figure 17 shows the format of UHL and UTL labels.

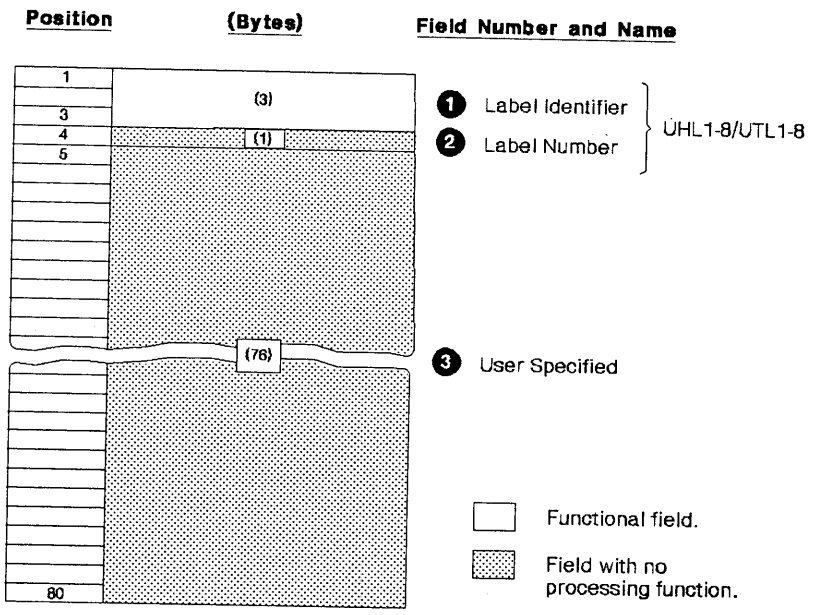



Figure 17. Format of Version 3 User Labels

**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. Data management accepts either UHL or UTL.

**2—Label Number** (1 byte)

- Contents: Any valid ISO/ANSI/FIPS character.
- Processing: This field is checked by the operating system for a valid ISO/ANSI/FIPS 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 ISO/ANSI/FIPS 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 ISO/ANSI/FIPS 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) 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 18 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.

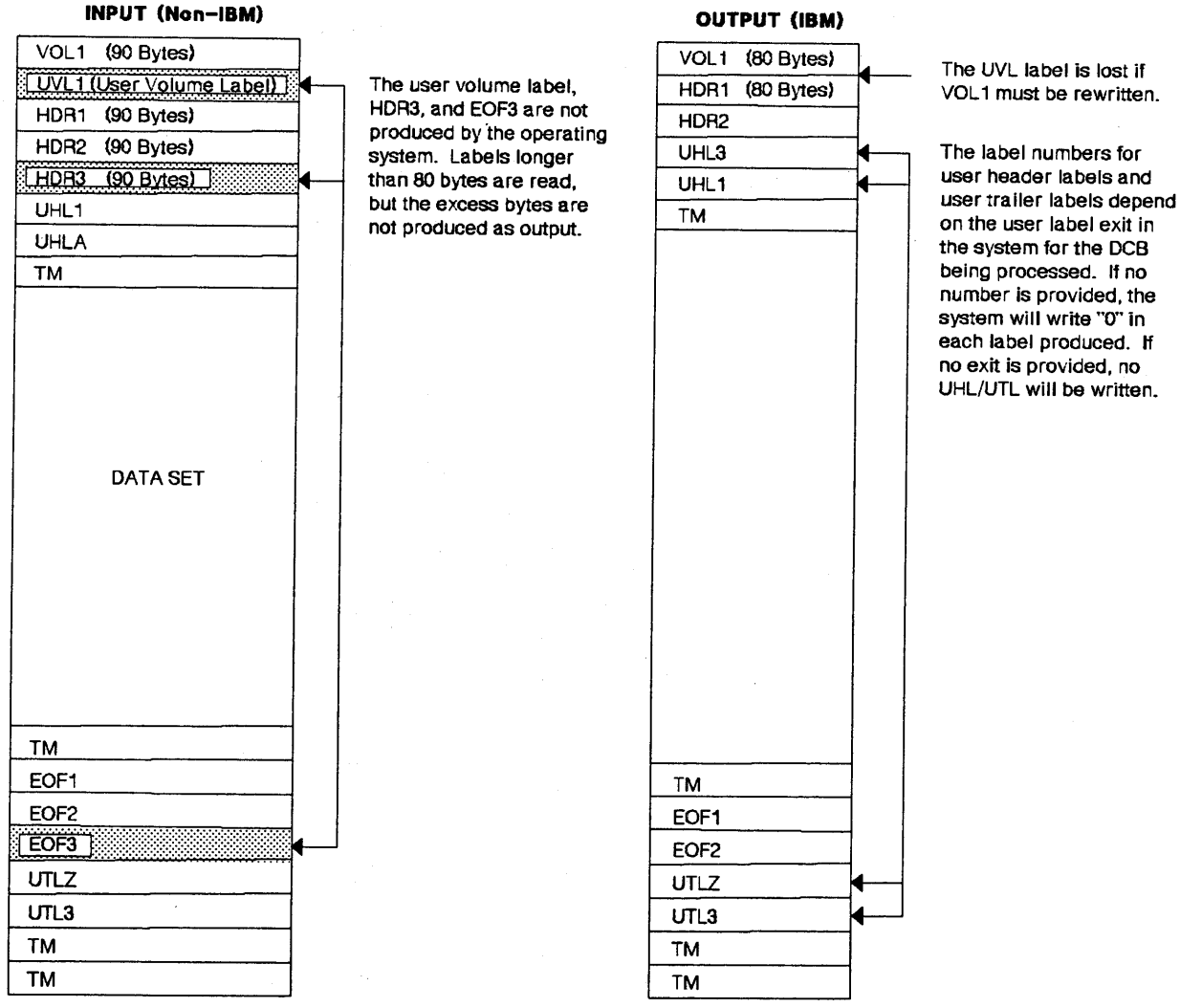



Figure 18. Example of an ISO/ANSI/FIPS Tape from a Non-IBM System and a Corresponding Output Tape

# Chapter 4. Nonstandard Labels

Product-Sensitive Programming Interface

This chapter contains product-sensitive programming interfaces provided by MVS/DFP. Installation exits and other product-sensitive interfaces are provided to allow the customer installation to perform tasks such as product tailoring, monitoring, modification, or diagnosis. They are dependent on the detailed design or implementation of the product. Such interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces may need to be changed in order to run with new product releases or versions, or as a result of service.

Nonstandard labels do not conform to the IBM or ISO/ANSI/FIPS standard label formats. They are labels which you design, and you provide routines to write and process them. There are no requirements as to the length, format, contents, and number of nonstandard labels, except that the first record on a BCD, EBCDIC, or ISCII/ASCII tape cannot be a standard volume label.

Nonstandard label routines may be inserted into the control program after system generation by link-editing them into LPALIB.

To insert your load modules into the SVC library during system generation, you use the SVCPARM macro instruction. With this macro instruction, you must specify the name of the partitioned data set and the names of members to be included in the SVC library. Member names for the first load module of each type of label processing routine are listed below. Member names for additional load modules must begin with the letters NSL or IGC. The format and specifications of the SVCPARM macro instruction are in *Initialization and Tuning*. For additional information on system generation, see *System Generation*.

| **Nonstandard Label Processing Routine** | **Control Program Routine** | **Member Name** |
|---|---|---|
| Input Header | Open | NSLOHDRI |
| | EOV | NLEHDRI |
| Output Header | Open | NSLOHDRO |
| | EOV | NSLEHDRO |
| Input Trailer | EOV | NSLETRLI |
| Output Trailer | EOV | NSLETRLO |
| | Close | NSLCTRLO |
| Restart Reader | Restart | NSLRHDRI |

If you use nonstandard tape labels and you want to use the dynamic device reconfiguration (DDR) option, you must perform your own volume verification. Note that you must be able to perform your verification within the first 48 bytes of any record in your nonstandard label.

Before system generation time, code a routine named NSLREPOS and link-edit it into a cataloged partitioned data set. Then, identify the member of the partitioned data set that contains NSLREPOS in the LPALIB system generation macro instruction. Link-edit NSLREPOS into the LPALIB after system generation.

For detailed information about these and other available exit routines, see *DFP: Customization*.

End of Product-Sensitive Programming Interface

# Chapter 5. Unlabeled Tapes

To process or create a tape with no labels, specify NL in the LABEL parameter of the DD statement. An unlabeled tape contains only data records and tape marks. The organization of data sets on one or more volumes is shown in Figure 19 on page 109. The data management routines of the operating system automatically write the tape marks on output and expect to find a similar placement of tape marks on input.

- A tape mark does not precede the first data set on any volume.
- A tape mark follows every data set.
- Two tape marks 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 tape mark preceding the first data set. In this case, the end-of-data-set condition is signaled by the reflective strip at the beginning of the tape.

Open/Close/EOV look ahead mounting does not accept any volume that is premounted and not verified on the next available unit (UCBNRY=0 and UCBYOLI=ZEROS). A demount message with a blank **vol ser** is issued.

**Note:** Automatic volume recognition (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, data management requests a demount of the incorrect volume and a mount of the specified volume. Eliminate specific requests in the JCL to avoid this problem.

## Bypass Label Processing (BLP) Option

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 processing is identical to NL processing, except that the check for an existing label is bypassed. BLP processing 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.

## Opening an Input Data Set

When you specify no labels, data management 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 data management directing the operator to mount the correct tape. The various error conditions that can occur during label verification are explained under Chapter 6, "Volume Label Verification and Volume Label Editor Routines" on page 115.

The search for a standard label is the only mount verification performed by data management. Without labels, neither the volume nor the data set can be positively identified and data management assumes that they are correct. The operator is responsible for checking 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, data management 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, you 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, the tape will be spaced through and removed from its reel.
- If you do not specify a sequence number, or specify zero, and the data set is not cataloged, data management 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 tape mark. If a tape mark 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 tape mark).

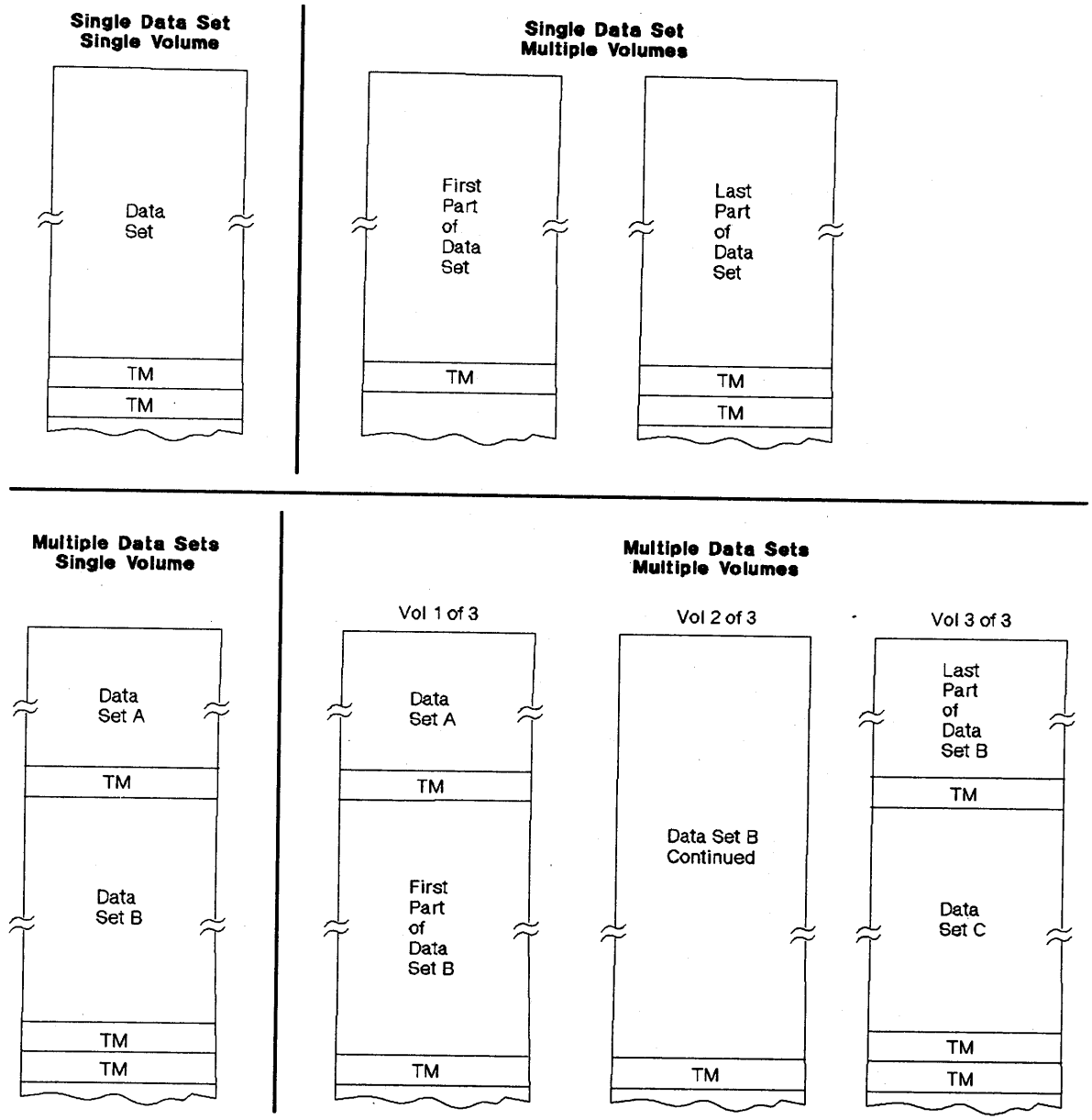



Figure 19. Organizations for Unlabeled Tapes

To position the tape, data management uses the requested data set sequence number shown 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 need not 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 may be open at any given time. If you attempt to begin processing a second data set on the same volume, processing is abnormally terminated.

### Read Backward

For the read backward (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, data management's EOV routine handles both end-of-data-set and end-of-volume conditions. These conditions occur when:

- A tape mark is read, or
- A force-end-of-volume (FEOV) macro instruction is executed by the processing program.

### Determining Volume Switch

The serial numbers of all volumes 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. Data management 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. Data management does not consider whether the data set on the current volume is followed by one or two tape marks. To determine whether additional volumes are required, data management maintains a volume sequence number in the data extent block (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, data management obtains the next volume serial number from the JFCB and switches volumes. Data management 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, data management 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, data management checks for the following conditions, in the order presented: (1) RACF authorization, (2) password protection, and (3) expiration date.

If the system-wide RACF tape protection option has been selected, data management checks the alter level authorization to the tape volume. If the tape volume is not defined to RACF, password protection is checked. If the tape volume is defined and the user is not authorized for ALTER, the tape is demounted; if the user is authorized for ALTER, password protection is bypassed.

If data management determines that the volume is password protected, a message to demount the tape is issued to the operator. Otherwise, data management continues processing by checking the expiration date. If the expiration date is earlier than the current date, 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, 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, data management requests that another volume be mounted. If the

operator accepts the tape, data management then destroys the standard label by writing a tape mark over it, thus providing you with the unlabeled tape you requested.

If the tape volume has been found to be RACF defined and the user is authorized for ALTER, that definition is deleted when the label is destroyed. If you do not want data management to perform this checking for you, you should insert your own label editor routine, as discussed under Chapter 6, "Volume Label Verification and Volume Label Editor Routines" on page 115. The various error conditions that can occur during verification of the first record are also explained under Chapter 6, "Volume Label Verification and Volume Label Editor Routines."

**CAUTION:**
**Doing multiple opens and closes without writing any user data in the area of the end-of-tape reflective marker may result in the header and trailer labels being written past the marker. Access methods detects the markers, but because the creation of empty data sets does not involve access methods, the end-of-tape marker will never be detected. This may cause the tape to run off the end of the reel.**

## Bypass Label Processing (BLP) Option

If you do not want data management to check an output tape for an existing standard label, you must specify BLP (instead of NL) in the LABEL parameter of the DD statement. In all other respects, tape processing under BLP is the same as if NL were specified. For the method of specifying BLP for the JES reader, see *Data Areas Volume 3* or *JES3 Initialization and Tuning*.

The BLP option is designed mainly to process blank (unused) tapes. You may want to write a tape mark, data, or a label on the blank tape. If BLP is not coded, data management will read through an entire blank tape looking for the first record.

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, data management makes 4 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 4 read operations by specifying BLP instead of NL.

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 jobclass; that is, the installation has the option to allow or disallow BLP by jobclass. It is recommended that BLP be disallowed completely, or if this is not possible, that one jobclass 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.

## 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.

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 Lxxxyy, 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 EOV 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, 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 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: (1) the tape will be spaced through and removed from its reel, or (2) 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, data management assumes that the data set is to be written as the first one on the volume.

To position the tape, data management maintains a logical data set sequence number in the unit control block (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 may be open at any given time. If you attempt to open a second data set on the same volume, processing is abnormally terminated.

## End-of-Volume on Output

Data management's EOV routine automatically switches volumes when an end-of-volume condition occurs on output; that is, when the reflective strip is encountered at the end of a tape or when an FEOV macro instruction is executed.

The EOV routine writes one tape mark after the data set on the current volume and checks the new volume to ensure that it 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, or INOUT) or when no output is written before closing (for OUTPUT or OUTIN), the close routine creates data set trailer labels.

## Restarting from a Checkpoint

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

1. Restores applicable control blocks to the conditions that existed when the checkpoint was taken.
2. Ensures that the first existing record on the tape is not a standard volume label (VOL1).
3. 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.
4. 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 zero 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. Volume Label Verification and Volume Label Editor Routines

Product-Sensitive Programming Interface

This chapter contains product-sensitive programming interfaces provided by MVS/DFP. Installation exits and other product-sensitive interfaces are provided to allow the customer installation to perform tasks such as product tailoring, monitoring, modification, or diagnosis. They are dependent on the detailed design or implementation of the product. Such interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces may need to be changed in order to run with new product releases or versions, or as a result of service.

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. For ISO/ANSI/FIPS tapes, the system checks for the correct version. If you specify that the tape has nonstandard labels or no labels, the system attempts to verify that the first record is not a standard volume label.

Because of conflicting label types or conflicting tape characteristics, various error conditions can occur during this 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 data management routine to create or destroy labels, as required. 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 EOV)

If either of your installation's editor routines consist 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 EOV routine. Transfer between the modules must be by name.

For detailed information about these and other available exit routines, see *DFP: Customization*.

End of Product-Sensitive Programming Interface

# Chapter 7. Using Tape Volumes Created by Other Systems

Occasionally, it may be necessary to process a tape volume that was 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 may be slightly or extremely different in its organization, its label formats, or its label contents. With the aid of this publication, a careful analysis of these factors will enable 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, it is recommended that you 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. Figures 7, 8, and 9 on pages 40, 44 and 50 show which fields of each label are functional for the operating system. 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, EOV1, 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. *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 job file control block after the DD statement is recorded there.

The operating system can obtain certain data set characteristics from the standard data set label 2 (HDR2/EOV2/EOF2). Some other 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, as long as 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 instruction 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 the first record on the tape is not identified as VOL1 and the data sets are followed by recognizable tape marks.

## 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 105.

If you want to ignore the nonstandard labels, you can retrieve the data set by treating the volume as an unlabeled tape. You 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 provided the data sets are followed by recognizable tape marks.

To position a tape at the desired data set, you must specify the correct data set sequence number in the DD statement. If a tape mark precedes the first data set and the LABEL subparameter LTM is specified, the system will test for and bypass, if present, a leading tape mark. If a tape mark 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 tape mark 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 tape mark, that is, a tape mark 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 or not it is preceded by a leading tape mark.

The specification of LTM in the LABEL parameter field does not make allowances for any other excess tape marks. You must make any such adjustments in the data set sequence number.

# Appendix A. Component Considerations

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 20 shows the component support for each type of label processing.

| Type | Assembler | Linkage Editor | Sort/Merge | Utilities | COBOL American National Standard V2 | COBOL American National Standard V3 | COBOL American National Standard V4 | FORTRAN | PL/I | RPG |
|---|---|---|---|---|---|---|---|---|---|---|
| Uses Data Management Facilities for Label Processing | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Supports Standard Labels (SL,AL) | Yes | Yes | Yes | SL-Yes<br>AL-IEHINITT | Yes | Yes | Yes | Yes | Yes | Yes |
| Supports Standard User Labels (SUL,AUL) | No | No | Yes | Yes | SUL-Yes<br>AUL-No | SUL-Yes<br>AUL-Yes | SUL-Yes<br>AUL-Yes | No<br>No | No<br>No | No<br>No |
| Supports Nonstandard Labels (NSL)[1] | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Supports Unlabeled Tape (NL) | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Supports Bypass Label Processing Option (BLP)[2] | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Supports Concatenated Data Sets with Unlike Attributes | No | Yes | No | No | No | No | No | No | No | No |

[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 specified at system generation, its use defaults to NL.

Figure 20. Component Support of Label Processing Types

# Appendix B. External Labels

External labels are affixed to the tape reels to provide visual identification of the volume and its contents. Normal tape volume control requires two 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 that contain checkpoint data sets.

To write on external labels, you should use an implement such as 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 21.

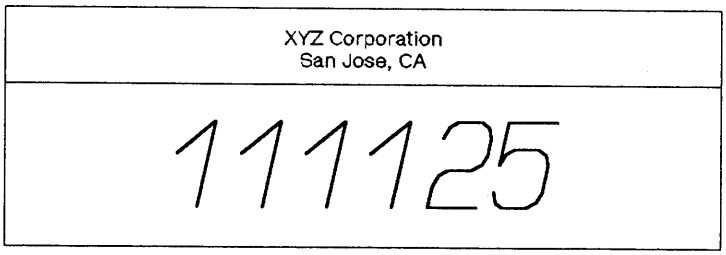



Figure 21. External Label for Reel Identification

## 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 22 on page 122.

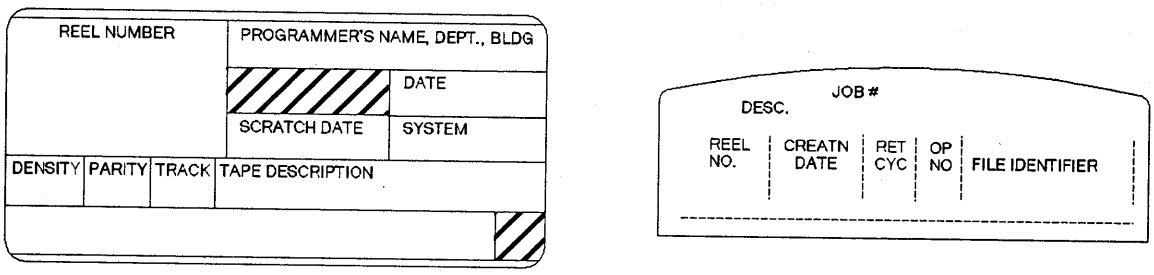



Figure 22. External Labels for Contents Identification

## 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 offline security of the checkpoint data set. For detailed information about the use of this label, see *Checkpoint/Restart User's Guide*.

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.

# Appendix C. Restart Work Areas

Product-Sensitive Programming Interface

This appendix contains product-sensitive programming interfaces provided by MVS/DFP. Installation exits and other product-sensitive interfaces are provided to allow the customer installation to perform tasks such as product tailoring, monitoring, modification, or diagnosis. They are dependent on the detailed design or implementation of the product. Such interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces may need to be changed in order to run with new product releases or versions, or as a result of service.

This appendix describes the restart table entry and the restart work and control block area. When your nonstandard label processing routine receives control from the control program's restart routine, register 9 contains the starting address of the table entry associated with the data set. The TABSEGAD field of the table entry points to the starting address of the work and control block area associated with the same data set.

## Table Entry

Figure 23 shows the format of the restart table entry. A description of each field follows.

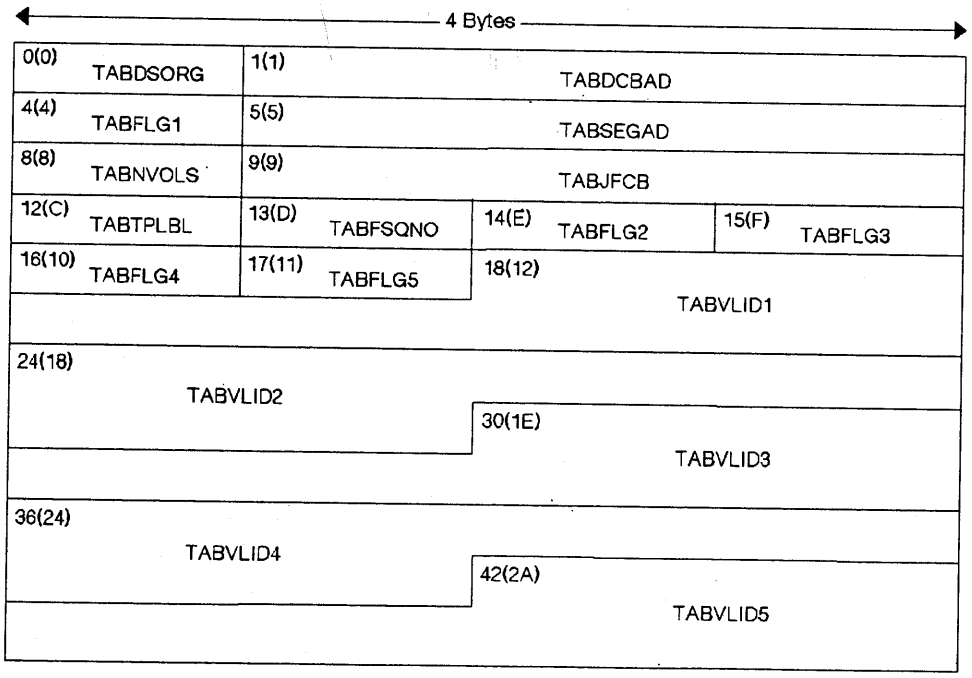



Figure 23. Restart Table Entry Format

| Offset | Field Name | Bytes | Description |
|---|---|---|---|
| 0(0) | TABDSORG | 1 | This field describes the data set organization used: |

| Bits | Settings | Meaning |
|---|---|---|
| 0 | 1 | Indexed sequential organization. |
| 1 | 1 | Physical sequential organization. |
| 2 | 1 | Direct organization. |
| 3-5 | | Reserved for future use. |
| 6 | 1 | Partitioned organization. |
| 7 | 1 | Unmovable—the data set contains location-dependent information. |

| | | | |
|---|---|---|---|
| 1(1) | TABDCBAD | 3 | Address of the DCB |
| 4(4) | TABFLG1 | 1 | This field contains the following information: |

| Bits | Settings | Meaning |
|---|---|---|
| 0 | 1 | Data set was specified in DD statement as NULLFILE or SYSCHECK. |
| 1 | 1 | Data set was specified in DD statement as SYSIN or SYSOUT. |
| 2 | 1 | Device type = direct access. |
| 3 | 1 | Device type = tape. |
| 4 | 1 | This is the last table entry in the restart table. |
| 7 | 1 | This is a DOS tape with an optional leading tape mark, and/or contains embedded DOS checkpoint records. |

| | | | |
|---|---|---|---|
| 5(5) | TABSEGAD | 3 | Address of the restart work and control block areafor this data set. |
| 8(8) | TABNVOLS | 1 | The total number of volumes for this data set, as specified in the DD statement. |
| 9(9) | TABJFCB | 3 | The relative track address (TTR) of the JFCB. |
| 12(C) | TABTPLBL | 1 | This field contains the following tape label information: |

| Bits | Settings | Meaning |
|---|---|---|
| 0 | 1 | I/O error in NSL processing. |
| 1 | | Reserved. |
| 2 | 1 | Bypass a leading tape mark, if present, on an unlabeled tape. Bit 7 is also set. |
| 3 | 1 | Bypass label processing. |
| 4 | 1 | ISO/ANSI/FIPS standard labels. |
| 5 | 1 | Nonstandard labels. |
| 6 | 1 | IBM standard labels. |
| 7 | 1 | No labels. |

| Offset | Field Name | Bytes | Description |
|---|---|---|---|
| 13(D) | TABFSQNO | 1 | Data set sequence number. |
| 14(E) | TABFLG2 | 1 | This field contains the following information:<br>**Bits** **Settings** **Meaning**<br>0 1 More than five volumes associated with this data set.<br>1 1 Partitioned organization concatenation.<br>2-7 Reserved |
| 15(F) | TABFLG3 | 1 | Reserved for possible future use. |
| 16(10) | TABFLG4 | 1 | Reserved for possible future use. |
| 17(11) | TABFLG5 | 1 | Reserved for possible future use. |
| 18(12) | TABVLID1 | 6 | The volume serial number of the first volume to be mounted for this data set. |
| 24(18) | TABVLID2 | 6 | The volume serial number of the second volume. |
| 30(1E) | TABVLID3 | 6 | The volume serial number of the third volume. |
| 36(24) | TABVLID4 | 6 | The volume serial number of the fourth volume. |
| 42(2A) | TABVLID5 | 6 | The volume serial number of the fifth volume. |

## Work and Control Block Area

Figure 24 shows the format of the restart work and control block area. A description of the control block fields follows.

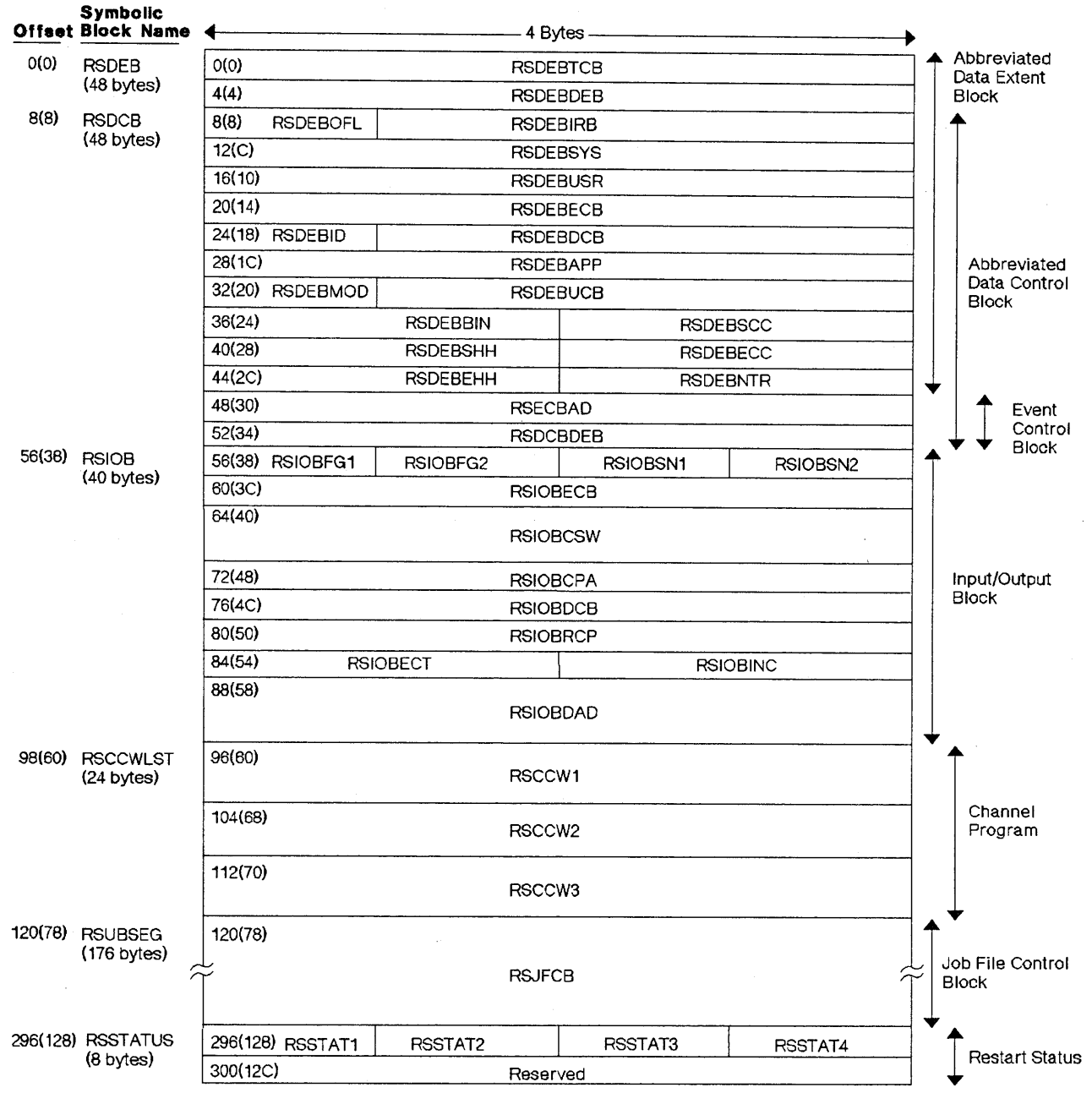



Figure 24. Restart Work and Control Block Area

| Offset | Field Name | Bytes | Description |
|---|---|---|---|
| 0(0) | RSDEBTCB | 4 | Address of TCB for this DEB. |
| 4(4) | RSDEBDEB | 4 | Address of the next DEB in the same task. |
| 8(8) | RSDEBOFL | 1 | Data set status flags. |
| 9(9) | RSDEBIRB | 3 | IRB address used for appendage asynchronous exits. |
| 12(C) | RSDEBSYS | 4 | Address of first IOB in the system purge chain. |
| 16(10) | RSDEBUSR | 4 | Address of first IOB in the user purge chain. |
| 20(14) | RSDEBECB | 4 | Address of a parameter list used to locate the purge ECB for an SVC purge request. |
| 24(18) | RSDEBID | 1 | A hexadecimal '0F' to identify this block as a DEB. |
| 25(19) | RSDEBDCB | 3 | Address of DEB associated with this DEB (RSDCB). |
| 28(1C) | RSDEBAPP | 4 | Address of the I/O appendage vector table. |
| 32(20) | RSDEBMOD | 1 | Device modified. |
| 33(21) | RSDEBUCB | 3 | Address of UCB. |
| 36(24) | RSDEBBIN | 2 | Bin number of direct access volume (data cell drive). |
| 38(26) | RSDEBSCC | 2 | Cylinder address for start of an extent limit. |
| 40(28) | RSDEBSHH | 2 | Track address for the start of an extent limit. |
| 42(2A) | RSDEBECC | 2 | Cylinder address for the end of an extent limit. |
| 44(2C) | RSDEBEHH | 2 | Track address for the end of an extent limit. |
| 46(2E) | RSDEBNTR | 2 | Number of tracks allocated to a given extent. |
| 48(30) | RSECBAD | 4 | Event control block (ECB). |
| 52(34) | RSDCBDEB | 4 | Address of DEB associated with this DCB (RSDEB). |
| 56(38) | RSIOBFG1 | 1 | Flag byte 1, as follows: |

| Bits | Settings | Meaning |
|---|---|---|
| 0-1 | 00 | No chaining. |
| | 01 | Command chaining. |
| | 10 | Data chaining. |
| | 11 | Both command and data chaining. |
| 2 | 1 | Error routine in control. |
| 3 | 1 | Device is to be repositioned. |
| 4 | 1 | Cyclic redundancy check (CRC) needed (tape only). |
| 5 | 1 | Exceptional condition (if this bit is on after the error routine returns, the error is considered permanent). |
| 6 | 1 | IOB unrelated flag (that is, nonsequential). |

| Offset | Field Name | Bytes | Description | | |
|---|---|---|---|---|---|
| | | | **Bits** | **Settings** | **Meaning** |
| | | | 7 | 0 | Error recovery procedure uses channel program address a START (IOB + 17). |
| | | | | 1 | RESTART error recovery procedure uses channel program address at IOBRESTR (IOB + 24). |
| 57(39) | RSIOBFG2 | 1 | Flag byte 2, as follows: | | |
| | | | **Bits** | **Settings** | **Meaning** |
| | | | 0 | 1 | Halt I/O has been issued. |
| | | | 1 | 1 | Sense will not be performed until the device is free. |
| | | | 2 | 1 | IOB has been purged. |
| | | | 3 | 1 | Home address (R0) record is to be read. |
| | | | 4-6 | | (variable) Internal I/O supervisor error correction flags. |
| | | | 7 | 1 | QSAM—error recovery in control for an IBM 2540 Card Read Punch with three buffers. |
| 58(3A) | RSIOBSN1 | 1 | First sense byte (device-dependent). | | |
| 59(3B) | RSIOBSN2 | 1 | Second sense byte (device-dependent). | | |
| 60(3C) | RSIOBECB | 4 | Address of the ECB to be posted (RSECBAD). | | |
| 64(40) | RSIOBCSW | 8 | CSW. | | |
| 72(48) | RSIOBCPA | 4 | Address of the channel program to be executed (RSCCW1). | | |
| 76(4C) | RSIOBDCB | 4 | Address of the DCB associated with this IOB (RSDCB). | | |
| 80(50) | RSIOBRCP | 4 | Restart address used by I/O supervisor error routines during error correction. | | |
| 84(54) | RSIOBECT | 2 | Value used to increase block count field in DCB for magnetic tape. | | |
| 86(56) | RSIOBINC | 2 | Used by I/O supervisor error routines to count temporary errors during retry. | | |
| 88(58) | RSIOBDAD | 8 | This field is used for direct access only. | | |
| 96(60) | RSCCW1 | 8 | Channel program area. | | |
| 104(68) | RSCCW2 | 8 | Channel program area. | | |
| 112(70) | RSCCW3 | 8 | Channel program area. | | |
| 120(78) | RSJFCB | 176 | Work area for job file control block. | | |
| 296(128) | RSSTAT1 | 1 | Status byte 1. | | |
| 297(129) | RSSTAT2 | 1 | Reserved for possible future use. | | |
| 298(12A) | RSSTAT3 | 1 | Reserved for possible future use. | | |

| Offset | Field Name | Bytes | Description |
|---|---|---|---|
| 299(12B) | RSSTAT4 | 1 | Reserved for possible future use. |
| 300(12C) | | 4 | Reserved for possible future use. |

End of Product-Sensitive Programming Interface

# Appendix D. ISO/ANSI/FIPS Version 3 Installation Exits

Product-Sensitive Programming Interface

This appendix contains product-sensitive programming interfaces provided by MVS/DFP. Installation exits and other product-sensitive interfaces are provided to allow the customer installation to perform tasks such as product tailoring, monitoring, modification, or diagnosis. They are dependent on the detailed design or implementation of the product. Such interfaces should be used only for these specialized purposes. Because of their dependencies on detailed design and implementation, it is to be expected that programs written to such interfaces may need to be changed in order to run with new product releases or versions, or as a result of service.

Four **installation exits** are provided, as defaults, for Version 3 volumes:

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

A fifth installation exit, WTOR, can be written (or modified, if one has already been written) by your installation to convert ISO/ANSI/FIPS 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. They execute in a privileged (supervisor) state and 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 may be modified to request continued processing. However, results are unpredictable in cases in which the label validation exit is entered and it has not been modified to also correct certain errors.

You can replace any of the IBM-supplied exit routines with your own, installation-written, exit routines.

For detailed information about these and other available exit routines, see *DFP: Customization*.

End of Product-Sensitive Programming Interface

# Appendix E. Equivalent ISCII/ASCII, EBCDIC, and Hollerith Codes

## ISCII/ASCII 7-Bit Code

The translate routine included in the system at system generation time translates ISCII/ASCII 7-bit code to and from EBCDIC.

All EBCDIC codes that translate to an ISCII/ASCII code that cannot be contained in seven bits are represented by the substitute character X'1A'.

## ISCII/ASCII 8-Bit Code

You may replace the IBM-supplied translation routine with an installation-written routine that translates ISCII/ASCII 8-bit code. Note, however, that such a modification will result in a volume that does not meet Version 3 standards, and therefore would require agreements between interchange parties.

The following tables show the relationship of EBCDIC and Hollerith code to ISCII/ASCII 8-bit code.

### EBCDIC to Hollerith and ISCII/ASCII

| EBCDIC (hex) | HOLLERITH (punches) | ISCII/ ASCII (hex) | EBCDIC (hex) | HOLLERITH (punches) | ISCII/ ASCII (hex) |
|---|---|---|---|---|---|
| 00 | 12-0-9-8-1 | 00 | 14 | 11-9-4 | 9D |
| 01 | 12-9-1 | 01 | 15 | 11-9-5 | 85 |
| 02 | 12-9-2 | 02 | 16 | 11-9-6 | 08 |
| 03 | 12-9-3 | 03 | 17 | 11-9-7 | 87 |
| 04 | 12-9-4 | 9C | 18 | 11-9-8 | 18 |
| 05 | 12-9-5 | 09 | 19 | 11-9-8-1 | 19 |
| 06 | 12-9-6 | 86 | 1A | 11-9-8-2 | 92 |
| 07 | 12-9-7 | 7F | 1B | 11-9-8-3 | 8F |
| 08 | 12-9-8 | 97 | 1C | 11-9-8-4 | 1C |
| 09 | 12-9-8-1 | 8D | 1D | 11-9-8-5 | 1D |
| 0A | 12-9-8-2 | 8E | 1E | 11-9-8-6 | 1E |
| 0B | 12-9-8-3 | 0B | 1F | 11-9-8-7 | 1F |
| 0C | 12-9-8-4 | 0C | 20 | 11-0-9-8-1 | 80 |
| 0D | 12-9-8-5 | 0D | 21 | 0-9-1 | 81 |
| 0E | 12-9-8-6 | 0E | 22 | 0-9-2 | 82 |
| 0F | 12-9-8-7 | 0F | 23 | 0-9-3 | 83 |
| 10 | 12-11-9-8-1 | 10 | 24 | 0-9-4 | 84 |
| 11 | 11-9-1 | 11 | 25 | 0-9-5 | 0A |
| 12 | 11-9-2 | 12 | 26 | 0-9-6 | 17 |
| 13 | 11-9-3 | 13 | 27 | 0-9-7 | 1B |

| EBCDIC (hex) | HOLLERITH (punches) | ISCII/ ASCII (hex) |
|---|---|---|
| 28 | 0-9-8 | 88 |
| 29 | 0-9-8-1 | 89 |
| 2A | 0-9-8-2 | 8A |
| 2B | 0-9-8-3 | 8B |
| 2C | 0-9-8-4 | 8C |
| 2D | 0-9-8-5 | 05 |
| 2E | 0-9-8-6 | 06 |
| 2F | 0-9-8-7 | 07 |
| 30 | 12-11-0-9-8-1 | 90 |
| 31 | 9-1 | 91 |
| 32 | 9-2 | 16 |
| 33 | 9-3 | 93 |
| 34 | 9-4 | 94 |
| 35 | 9-5 | 95 |
| 36 | 9-6 | 96 |
| 37 | 9-7 | 04 |
| 38 | 9-8 | 98 |
| 39 | 9-8-1 | 99 |
| 3A | 9-8-2 | 9A |
| 3B | 9-8-3 | 9B |
| 3C | 9-8-4 | 14 |
| 3D | 9-8-5 | 15 |
| 3E | 9-8-6 | 9E |
| 3F | 9-8-7 | 1A |
| 40 | (none) | 20 |
| 41 | 12-0-9-1 | A0 |
| 42 | 12-0-9-2 | A1 |
| 43 | 12-0-9-3 | A2 |
| 44 | 12-0-9-4 | A3 |
| 45 | 12-0-9-5 | A4 |
| 46 | 12-0-9-6 | A5 |
| 47 | 12-0-9-7 | A6 |
| 48 | 12-0-9-8 | A7 |
| 49 | 12-8-1 | A8 |
| 4A | 12-8-2 | 5B |
| 4B | 12-8-3 | 2E |
| 4C | 12-8-4 | 3C |
| 4D | 12-8-5 | 28 |
| 4E | 12-8-6 | 2B |
| 4F | 12-8-7 | 21 |
| 50 | 12 | 26 |
| 51 | 12-11-9-1 | A9 |
| 52 | 12-11-9-2 | AA |
| 53 | 12-11-9-3 | AB |
| 54 | 12-11-9-4 | AC |
| 55 | 12-11-9-5 | AD |
| 56 | 12-11-9-6 | AE |
| 57 | 12-11-9-7 | AF |
| 58 | 12-11-9-8 | B0 |
| 59 | 11-8-1 | B1 |
| 5A | 11-8-2 | 5D |
| 5B | 11-8-3 | 24 |
| 5C | 11-8-4 | 2A |
| 5D | 11-8-5 | 29 |
| 5E | 11-8-6 | 3B |
| 5F | 11-8-7 | 5E |
| 60 | 11 | 2D |
| 61 | 0-1 | 2F |
| 62 | 11-0-9-2 | B2 |
| 63 | 11-0-9-3 | B3 |
| 64 | 11-0-9-4 | B4 |
| 65 | 11-0-9-5 | B5 |
| 66 | 11-0-9-6 | B6 |
| 67 | 11-0-9-7 | B7 |
| 68 | 11-0-9-8 | B8 |
| 69 | 0-8-1 | B9 |
| 6A | 12-11 | 7C |
| 6B | 0-8-3 | 2C |
| 6C | 0-8-4 | 25 |
| 6D | 0-8-5 | 5F |
| 6E | 0-8-6 | 3E |
| 6F | 0-8-7 | 3F |
| 70 | 12-11-0 | BA |
| 71 | 12-11-0-9-1 | BB |
| 72 | 12-11-0-9-2 | BC |
| 73 | 12-11-0-9-3 | BD |
| 74 | 12-11-0-9-4 | BE |
| 75 | 12-11-0-9-5 | BF |
| 76 | 12-11-0-9-6 | C0 |
| 77 | 12-11-0-9-7 | C1 |
| 78 | 12-11-0-9-8 | C2 |
| 79 | 8-1 | 60 |
| 7A | 8-2 | 3A |
| 7B | 8-3 | 23 |
| 7C | 8-4 | 40 |
| 7D | 8-5 | 27 |
| 7E | 8-6 | 3D |
| 7F | 8-7 | 22 |
| 80 | 12-0-8-1 | C3 |
| 81 | 12-0-1 | 61 |

| EBCDIC (hex) | HOLLERITH (punches) | ISCII/ ASCII (hex) |
|---|---|---|
| 82 | 12-0-2 | 62 |
| 83 | 12-0-3 | 63 |
| 84 | 12-0-4 | 64 |
| 85 | 12-0-5 | 65 |
| 86 | 12-0-6 | 66 |
| 87 | 12-0-7 | 67 |
| 88 | 12-0-8 | 68 |
| 89 | 12-0-9 | 69 |
| 8A | 12-0-8-2 | C4 |
| 8B | 12-0-8-3 | C5 |
| 8C | 12-0-8-4 | C6 |
| 8D | 12-0-8-5 | C7 |
| 8E | 12-0-8-6 | C8 |
| 8F | 12-0-8-7 | C9 |
| 90 | 12-11-8-1 | CA |
| 91 | 12-11-1 | 6A |
| 92 | 12-11-2 | 6B |
| 93 | 12-11-3 | 6C |
| 94 | 12-11-4 | 6D |
| 95 | 12-11-5 | 6E |
| 96 | 12-11-6 | 6F |
| 97 | 12-11-7 | 70 |
| 98 | 12-11-8 | 71 |
| 99 | 12-11-9 | 72 |
| 9A | 12-11-8-2 | CB |
| 9B | 12-11-8-3 | CC |
| 9C | 12-11-8-4 | CD |
| 9D | 12-11-8-5 | CE |
| 9E | 12-11-8-6 | CF |
| 9F | 12-11-8-7 | D0 |
| A0 | 11-0-8-1 | D1 |
| A1 | 11-0-1 | 7E |
| A2 | 11-0-2 | 73 |
| A3 | 11-0-3 | 74 |
| A4 | 11-0-4 | 75 |
| A5 | 11-0-5 | 76 |
| A6 | 11-0-6 | 77 |
| A7 | 11-0-7 | 78 |
| A8 | 11-0-8 | 79 |
| A9 | 11-0-9 | 7A |
| AA | 11-0-8-2 | D2 |
| AB | 11-0-8-3 | D3 |
| AC | 11-0-8-4 | D4 |
| AD | 11-0-8-5 | D5 |
| AE | 11-0-8-6 | D6 |
| AF | 11-0-8-7 | D7 |
| B0 | 12-11-0-8-1 | D8 |
| B1 | 12-11-0-1 | D9 |
| B2 | 12-11-0-2 | DA |
| B3 | 12-11-0-3 | DB |
| B4 | 12-11-0-4 | DC |
| B5 | 12-11-0-5 | DD |
| B6 | 12-11-0-6 | DE |
| B7 | 12-11-0-7 | DF |
| B8 | 12-11-0-8 | E0 |
| B9 | 12-11-0-9 | E1 |
| BA | 12-11-0-8-2 | E2 |
| BB | 12-11-0-8-3 | E3 |
| BC | 12-11-0-8-4 | E4 |
| BD | 12-11-0-8-5 | E5 |
| BE | 12-11-0-8-6 | E6 |
| BF | 12-11-0-8-7 | E7 |
| C0 | 12-0 | 7B |
| C1 | 12-1 | 41 |
| C2 | 12-2 | 42 |
| C3 | 12-3 | 43 |
| C4 | 12-4 | 44 |
| C5 | 12-5 | 45 |
| C6 | 12-6 | 46 |
| C7 | 12-7 | 47 |
| C8 | 12-8 | 48 |
| C9 | 12-9 | 49 |
| CA | 12-0-9-8-2 | E8 |
| CB | 12-0-9-8-3 | E9 |
| CC | 12-0-9-8-4 | EA |
| CD | 12-0-9-8-5 | EB |
| CE | 12-0-9-8-6 | EC |
| CF | 12-0-9-8-7 | ED |
| D0 | 11-0 | 7D |
| D1 | 11-1 | 4A |
| D2 | 11-2 | 4B |
| D3 | 11-3 | 4C |
| D4 | 11-4 | 4D |
| D5 | 11-5 | 4E |
| D6 | 11-6 | 4F |
| D7 | 11-7 | 50 |
| D8 | 11-8 | 51 |
| D9 | 11-9 | 52 |
| DA | 12-11-9-8-2 | EE |
| DB | 12-11-9-8-3 | EF |

| EBCDIC (hex) | HOLLERITH (punches) | ISCII/ ASCII (hex) |
|---|---|---|
| DC | 12-11-9-8-4 | F0 |
| DD | 12-11-9-8-5 | F1 |
| DE | 12-11-9-8-6 | F2 |
| DF | 12-11-9-8-7 | F3 |
| E0 | 0-8-2 | 5C |
| E1 | 11-0-9-1 | 9F |
| E2 | 0-2 | 53 |
| E3 | 0-3 | 54 |
| E4 | 0-4 | 55 |
| E5 | 0-5 | 56 |
| E6 | 0-6 | 57 |
| E7 | 0-7 | 58 |
| E8 | 0-8 | 59 |
| E9 | 0-9 | 5A |
| EA | 11-0-9-8-2 | F4 |
| EB | 11-0-9-8-3 | F5 |
| EC | 11-0-9-8-4 | F6 |
| ED | 11-0-9-8-5 | F7 |
| EE | 11-0-9-8-6 | F8 |
| EF | 11-0-9-8-7 | F9 |
| F0 | 0 | 30 |
| F1 | 1 | 31 |
| F2 | 2 | 32 |
| F3 | 3 | 33 |
| F4 | 4 | 34 |
| F5 | 5 | 35 |
| F6 | 6 | 36 |
| F7 | 7 | 37 |
| F8 | 8 | 38 |
| F9 | 9 | 39 |
| FA | 12-11-0-9-8-2 | FA |
| FB | 12-11-0-9-8-3 | FB |
| FC | 12-11-0-9-8-4 | FC |
| FD | 12-11-0-9-8-5 | FD |
| FE | 12-11-0-9-8-6 | FE |
| FF | 12-11-0-9-8-7 | FF |

## ISCII/ASCII to EBCDIC and Hollerith

| ISCII/ ASCII (hex) | HOLLERITH (punches) | EBCDIC (hex) |
|---|---|---|
| 00 | 12-0-9-8-1 | 00 |
| 01 | 12-9-1 | 01 |
| 02 | 12-9-2 | 02 |
| 03 | 12-9-3 | 03 |
| 04 | 9-7 | 37 |
| 05 | 0-9-8-5 | 2D |
| 06 | 0-9-8-6 | 2E |
| 07 | 0-9-8-7 | 2F |
| 08 | 11-9-6 | 16 |
| 09 | 12-9-5 | 05 |
| 0A | 0-9-5 | 25 |
| 0B | 12-9-8-3 | 0B |
| 0C | 12-9-8-4 | 0C |
| 0D | 12-9-8-5 | 0D |
| 0E | 12-9-8-6 | 0E |
| 0F | 12-9-8-7 | 0F |
| 10 | 12-11-9-8-1 | 10 |
| 11 | 11-9-1 | 11 |
| 12 | 11-9-2 | 12 |
| 13 | 11-9-3 | 13 |
| 14 | 9-8-4 | 3C |
| 15 | 9-8-5 | 3D |
| 16 | 9-2 | 32 |
| 17 | 0-9-6 | 26 |
| 18 | 11-9-8 | 18 |
| 19 | 11-9-8-1 | 19 |
| 1A | 9-8-7 | 3F |
| 1B | 0-9-7 | 27 |
| 1C | 11-9-8-4 | 1C |
| 1D | 11-9-8-5 | 1D |
| 1E | 11-9-8-6 | 1E |
| 1F | 11-9-8-7 | 1F |
| 20 | (none) | 40 |
| 21 | 12-8-7 | 4F |
| 22 | 8-7 | 7F |
| 23 | 8-3 | 7B |
| 24 | 11-8-3 | 5B |
| 25 | 0-8-4 | 6C |
| 26 | 12 | 50 |
| 27 | 8-5 | 7D |
| 28 | 12-8-5 | 4D |
| 29 | 11-8-5 | 5D |
| 2A | 11-8-4 | 5C |
| 2B | 12-8-6 | 4E |
| 2C | 0-8-3 | 6B |
| 2D | 11 | 60 |
| 2E | 12-8-3 | 4B |
| 2F | 0-1 | 61 |
| 30 | 0 | F0 |
| 31 | 1 | F1 |
| 32 | 2 | F2 |
| 33 | 3 | F3 |
| 34 | 4 | F4 |
| 35 | 5 | F5 |
| 36 | 6 | F6 |
| 37 | 7 | F7 |
| 38 | 8 | F8 |
| 39 | 9 | F9 |
| 3A | 8-2 | 7A |
| 3B | 11-8-6 | 5E |
| 3C | 12-8-4 | 4C |
| 3D | 8-6 | 7E |
| 3E | 0-8-6 | 6E |
| 3F | 0-8-7 | 6F |
| 40 | 8-4 | 7C |
| 41 | 12-1 | C1 |
| 42 | 12-2 | C2 |
| 43 | 12-3 | C3 |
| 44 | 12-4 | C4 |
| 45 | 12-5 | C5 |
| 46 | 12-6 | C6 |
| 47 | 12-7 | C7 |
| 48 | 12-8 | C8 |
| 49 | 12-9 | C9 |
| 4A | 11-1 | D1 |
| 4B | 11-2 | D2 |
| 4C | 11-3 | D3 |
| 4D | 11-4 | D4 |
| 4E | 11-5 | D5 |
| 4F | 11-6 | D6 |
| 50 | 11-7 | D7 |
| 51 | 11-8 | D8 |
| 52 | 11-9 | D9 |
| 53 | 0-2 | E2 |
| 54 | 0-3 | E3 |
| 55 | 0-4 | E4 |
| 56 | 0-5 | E5 |
| 57 | 0-6 | E6 |
| 58 | 0-7 | E7 |
| 59 | 0-8 | E8 |

| ISCII/ ASCII (hex) | HOLLERITH (punches) | EBCDIC (hex) |
|---|---|---|
| 5A | 0-9 | E9 |
| 5B | 12-8-2 | 4A |
| 5C | 0-8-2 | 50 |
| 5D | 11-8-2 | 5A |
| 5E | 11-8-7 | 5F |
| 5F | 0-8-5 | 6D |
| 60 | 8-1 | 79 |
| 61 | 12-0-1 | 81 |
| 62 | 12-0-2 | 82 |
| 63 | 12-0-3 | 83 |
| 64 | 12-0-4 | 84 |
| 65 | 12-0-5 | 85 |
| 66 | 12-0-6 | 86 |
| 67 | 12-0-7 | 87 |
| 68 | 12-0-8 | 88 |
| 69 | 12-0-9 | 89 |
| 6A | 12-11-1 | 91 |
| 6B | 12-11-2 | 92 |
| 6C | 12-11-3 | 93 |
| 6D | 12-11-4 | 94 |
| 6E | 12-11-5 | 95 |
| 6F | 12-11-6 | 96 |
| 70 | 12-11-7 | 97 |
| 71 | 12-11-8 | 98 |
| 72 | 12-11-9 | 99 |
| 73 | 11-0-2 | A2 |
| 74 | 11-0-3 | A3 |
| 75 | 11-0-4 | A4 |
| 76 | 11-0-5 | A5 |
| 77 | 11-0-6 | A6 |
| 78 | 11-0-7 | A7 |
| 79 | 11-0-8 | A8 |
| 7A | 11-0-9 | A9 |
| 7B | 12-0 | C0 |
| 7C | 12-11 | 6A |
| 7D | 11-0 | D0 |
| 7E | 11-0-1 | A1 |
| 7F | 12-9-7 | 07 |
| 80 | 11-0-9-8-1 | 20 |
| 81 | 0-9-1 | 21 |
| 82 | 0-9-2 | 22 |
| 83 | 0-9-3 | 23 |
| 84 | 0-9-4 | 24 |
| 85 | 11-9-5 | 15 |
| 86 | 12-9-6 | 06 |

| ISCII/ ASCII (hex) | HOLLERITH (punches) | EBCDIC (hex) |
|---|---|---|
| 87 | 11-9-7 | 17 |
| 88 | 0-9-8 | 28 |
| 89 | 0-9-8-1 | 29 |
| 8A | 0-9-8-2 | 2A |
| 8B | 0-9-8-3 | 2B |
| 8C | 0-9-8-4 | 2C |
| 8D | 12-9-8-1 | 09 |
| 8E | 12-9-8-2 | 0A |
| 8F | 11-9-8-3 | 1B |
| 90 | 12-11-0-9-8-1 | 30 |
| 91 | 9-1 | 31 |
| 92 | 11-9-8-2 | 1A |
| 93 | 9-3 | 33 |
| 94 | 9-4 | 34 |
| 95 | 9-5 | 35 |
| 96 | 9-6 | 36 |
| 97 | 12-9-8 | 08 |
| 98 | 9-8 | 38 |
| 99 | 9-8-1 | 39 |
| 9A | 9-8-2 | 3A |
| 9B | 9-8-3 | 3B |
| 9C | 12-9-4 | 04 |
| 9D | 11-9-4 | 14 |
| 9E | 9-8-6 | 3E |
| 9F | 11-0-9-1 | E1 |
| A0 | 12-0-9-1 | 41 |
| A1 | 12-0-9-2 | 42 |
| A2 | 12-0-9-3 | 43 |
| A3 | 12-0-9-4 | 44 |
| A4 | 12-0-9-5 | 45 |
| A5 | 12-0-9-6 | 46 |
| A6 | 12-0-9-7 | 47 |
| A7 | 12-0-9-8 | 48 |
| A8 | 12-8-1 | 49 |
| A9 | 12-11-9-1 | 51 |
| AA | 12-11-9-2 | 52 |
| AB | 12-11-9-3 | 53 |
| AC | 12-11-9-4 | 54 |
| AD | 12-11-9-5 | 55 |
| AE | 12-11-9-6 | 56 |
| AF | 12-11-9-7 | 57 |
| B0 | 12-11-9-8 | 58 |
| B1 | 11-8-1 | 59 |
| B2 | 11-0-9-2 | 62 |
| B3 | 11-0-9-3 | 63 |

| ISCII/ ASCII (hex) | HOLLERITH (punches) | EBCDIC (hex) |
|---|---|---|
| B4 | 11-0-9-4 | 64 |
| B5 | 11-0-9-5 | 65 |
| B6 | 11-0-9-6 | 66 |
| B7 | 11-0-9-7 | 67 |
| B8 | 11-0-9-8 | 68 |
| B9 | 0-8-1 | 69 |
| BA | 12-11-0 | 70 |
| BB | 12-11-0-9-1 | 71 |
| BC | 12-11-0-9-2 | 72 |
| BD | 12-11-0-9-3 | 73 |
| BE | 12-11-0-9-4 | 74 |
| BF | 12-11-0-9-5 | 75 |
| C0 | 12-11-0-9-6 | 76 |
| C1 | 12-11-0-9-7 | 77 |
| C2 | 12-11-0-9-8 | 78 |
| C3 | 12-0-8-1 | 80 |
| C4 | 12-0-8-2 | 8A |
| C5 | 12-0-8-3 | 8B |
| C6 | 12-0-8-4 | 8C |
| C7 | 12-0-8-5 | 8D |
| C8 | 12-0-8-6 | 8E |
| C9 | 12-0-8-7 | 8F |
| CA | 12-11-8-1 | 90 |
| CB | 12-11-8-2 | 9A |
| CC | 12-11-8-3 | 9B |
| CD | 12-11-8-4 | 9C |
| CE | 12-11-8-5 | 9D |
| CF | 12-11-8-6 | 9E |
| D0 | 12-11-8-7 | 9F |
| D1 | 11-0-8-1 | A0 |
| D2 | 11-0-8-2 | AA |
| D3 | 11-0-8-3 | AB |
| D4 | 11-0-8-4 | AC |
| D5 | 11-0-8-5 | AD |
| D6 | 11-0-8-6 | AE |
| D7 | 11-0-8-7 | AF |
| D8 | 12-11-0-8-1 | B0 |
| D9 | 12-11-0-1 | B1 |
| DA | 12-11-0-2 | B2 |
| DB | 12-11-0-3 | B3 |
| DC | 12-11-0-4 | B4 |
| DD | 12-11-0-5 | B5 |
| DE | 12-11-0-6 | B6 |
| DF | 12-11-0-7 | B7 |
| E0 | 12-11-0-8 | B8 |
| E1 | 12-11-0-9 | B9 |
| E2 | 12-11-0-8-2 | BA |
| E3 | 12-11-0-8-3 | BB |
| E4 | 12-11-0-8-4 | BC |
| E5 | 12-11-0-8-5 | BD |
| E6 | 12-11-0-8-6 | BE |
| E7 | 12-11-0-8-7 | BF |
| E8 | 12-0-9-8-2 | CA |
| E9 | 12-0-9-8-3 | CB |
| EA | 12-0-9-8-4 | CC |
| EB | 12-0-9-8-5 | CD |
| EC | 12-0-9-8-6 | CE |
| ED | 12-0-9-8-7 | CF |
| EE | 12-11-9-8-2 | DA |
| EF | 12-11-9-8-3 | DB |
| F0 | 12-11-9-8-4 | DC |
| F1 | 12-11-9-8-5 | DD |
| F2 | 12-11-9-8-6 | DE |
| F3 | 12-11-9-8-7 | DF |
| F4 | 11-0-9-8-2 | EA |
| F5 | 11-0-9-8-3 | EB |
| F6 | 11-0-9-8-4 | EC |
| F7 | 11-0-9-8-5 | ED |
| F8 | 11-0-9-8-6 | EE |
| F9 | 11-0-9-8-7 | EF |
| FA | 12-11-0-9-8-2 | FA |
| FB | 12-11-0-9-8-3 | FB |
| FC | 12-11-0-9-8-4 | FC |
| FD | 12-11-0-9-8-5 | FD |
| FE | 12-11-0-9-8-6 | FE |
| FF | 12-11-0-9-8-7 | FF |

## Translation Irregularities

Six irregularities exist in the graphic character set between ISCII/ASCII 7-bit code, ISCII/ASCII 8-bit code, and EBCDIC. These are:

| EBCDIC Code (Graphic) | (Hex) | Bit Code (Graphic) | (Hex) | 7-Bit Code (Graphic) | (Hex) |
|---|---|---|---|---|---|
| ¢ | 4A | [ | 5B | ] | 5B |
| ! | 5A | ] | 5D | ] | 5D |
| [ | AD | (n/a) | D5 | Sub | 1A |
| ] | BD | (n/a) | E5 | Sub | 1A |
| \| | 4F | ! | 21 | ! | 21 |
| \| | 6A | \| | 7C | \| | 7C |

Thus, for example, an exclamation mark is coded in EBCDIC as X'5A', but in ISCII/ASCII 7-bit and 8-bit codes as X'21'.

# Abbreviations

The following terms and abbreviations are defined as they are used in the MVS/DFP library. If you do find the term or abbreviation you are looking for, see *Dictionary of Computing*, SC20-1699.

This list includes acronyms and abbreviations developed by the American National Standards Institute (ANSI) and the International Organization for Standardization (ISO). This material is reproduced from the *American National Dictionary for Information Processing*, copyright 1977 by the Computer and Business Equipment Manufacturers American National Standards Institute, 1430 Broadway, New York, New York 10018.

## A

**ACB.** Access method control block.

**ACDS.** Active control data set.

**ACS.** Automatic class selection.

**ADDR.** Addressed processing.

**AL.** American National Standard Labels.

**ANSI.** American National Standards Institute.

**APF.** Authorized program facility.

**ASCII.** American National Standard Code for Information Interchange.

**ASI.** Asynchronous interrupt.

**AUL.** ANSI and user header or trailer labels.

**AVR.** Automatic volume recognition.

## B

**BCD.** Binary coded decimal

**BCDIC.** Binary coded decimal interchange code.

**BDAM.** Basic direct access method.

**BISAM.** Basic indexed sequential access method.

**BLP.** Bypass label processing.

**BPAM.** Basic partitioned access method.

**bpi.** Bits per inch.

**BPI.** Bytes per inch.

**BSAM.** Basic sequential access method.

## C

**CCW.** Channel command word.

**CI.** Control interval.

**CKDS.** Cryptographic key data set.

**COMMDS.** Communications data set.

**CSECT.** Control section.

**CSW.** Channel status word.

**CVOL.** Control volume.

**CVT.** Communication vector table.

## D

**DCB.** Data control block.

**DCBD.** Data control block dummy section.

**DD.** Data definition.

**DDNAME.** Data definition name.

**DEB.** Data extent block.

**DECB.** Data event control block.

**DES.** Data Encryption Standard.

**DFP.** Data Facility Product.

**DOS.** Disk operating system.

**DSCB.** Data set control block.

**DSECT.** Dummy control section.

**DSL.** Data extent block save list.

**DSNAME.** Data set name.

## E

**EBCDIC.** Extended binary-coded decimal interchange code.

**ECB.** Event control block.

**EDT.** Eligible device table.

**EOD.** End-of-data.

**EOF.** End-of-file.

**EOM.** End-of-module.

**EOV.** End-of-volume.

**ERP.** Error recovery procedure.

**EXCP.** Execute channel program.

## F

**FIPS.** Federal Information Processing Standard.

## G

**GCR.** Group coded recording.

**GDG.** Generation data group.

**GDGNT.** Generation data group name table.

**GDS.** Generation data set.

**GTF.** Generalized trace facility.

## H

**HDR.** Header label.

## I

**I/O.** Input/output.

**IOB.** Input/output block.

**IOGEN.** I/O device generation.

**IOS.** Input/output supervisor.

**IPL.** Initial program load.

**IRB.** I/O restart block.

**IRG.** Interrecord gap.

**ISAM.** Indexed sequential access method.

**ISCII.** International Standard Code for Information Interchange.

**ISO.** International Organization for Standardization.

## J

**JCL.** Job control language.

**JES.** Job entry subsystem.

**JFCB.** Job file control block.

## K

**K.** Kilobyte

## L

**LDS.** Linear data set.

**LRI.** Logical record interface.

## M

**MLA.** Multilevel alias facility.

**MSS.** Mass Storage System.

**MSVC.** Mass Storage Volume Control.

**MVS.** Multiple virtual storage.

## N

**NIP.** Nucleus initialization program.

**NL.** No label.

**NSL.** Nonstandard label.

**NUP.** No update.

**NVR.** Non-VSAM volume record.

## O

**O/C/EOV.** Open/close/end-of-volume.

**OS.** Operating system.

**OS CVOL.** Operating system control volume.

**OS/VS.** Operating system/virtual storage.

## P

**PCI.** Program-controlled interruption.

**PDS.** Partitioned data set.

**PSW.** Program status word.

## Q

**QISAM.** Queued indexed sequential access method.

**QSAM.** Queued sequential access method.

## R

**RACF.** Resource Access Control Facility.

**RBA.** Relative byte address.

**RDBACK.** Read backward.

**RPL.** Request parameter list.

**RPG.** Report Program Generator.

**RTM.** Recovery/termination manager.

**RTN.** Routine.

## S

**SAM.** Sequential access method.

**SCDS.** Source control data set.

**SKP.** Skip sequential processing.

**SL.** IBM standard label.

**SMAS.** Storage management address space.

**SMF.** System management facilities.

**SML.** Storage Management Library.

**SMS.** Storage Management Subsystem.

**SSL.** Storage Subsystem Library.

**SVC.** Supervisor call.

**SVCLIB.** Supervisor call library.

**SVRB.** Supervisor request block.

**SYSGEN.** System generation.

**SYSIN.** System input stream.

**SYSOUT.** System output stream.

## T

**TCAM.** Telecommunications access method.

**TCB.** Task control block.

**TIOT.** Task I/O table.

**TSO.** Time sharing option.

**TTR.** Track record address.

## U

**UCB.** Unit control block.

**UCS.** Universal character set.

**UHL.** User header label.

**UPD.** Update mode.

**USAR.** User security authorization record.

**USVR.** User security verification routine.

**UTL.** User trailer label.

## V

**VICE.** Volume index control entry.

**VIO.** Virtual input/output.

**VSAM.** Virtual storage access method.

**VTAM.** Virtual telecommunications access method.

## W

**WTG.** Where-to-go table.

## X

**XA.** Extended architecture.

# Glossary

The following terms and abbreviations are defined as they are used in the MVS/DFP library. If you do find the term or abbreviation you are looking for, see *Dictionary of Computing*, SC20-1699.

This list includes acronyms and abbreviations developed by the American National Standards Institute (ANSI) and the International Organization for Standardization (ISO). This material is reproduced from the *American National Dictionary for Information Processing*, copyright 1977 by the Computer and Business Equipment Manufacturers American National Standards Institute, 1430 Broadway, New York, New York 10018.

## A

**abnormal end (ABEND).** Termination of a task prior to its completion as a result of an error condition that could not be resolved by error recovery facilities during task execution.

**access method.** A technique for organizing and moving data between main storage and input/output devices.

**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.

**authorized program facility (APF).** A system facility that permits the identification of programs that are authorized to use restriced functions.

**automatic data protection (ADSP).** A user attribute that causes all permanent DASD data sets defined by the user to be automatically defined to RACF.

**automatic volume recognition (AVR).** A part of the job scheduler that allows an operator to premount volumes on unused drives for assignment to later job steps.

**auxiliary storage.** All addressable storage, other than the memory of a processing unit, that can be accessed by means of an input/output channel; for example, storage on DASD, tape, or mass storage system volumes.

**availability.** For a storage subsystem, the degree to which a data set can be accessed when requested by a user.

## B

**backup.** The process of copying data and storing it for use in case the original data is somehow damaged or destroyed.

**backup data set.** A copy that can be used to replace or reconstruct a damaged data set.

**base RBA.** In VSAM, the relative byte address (RBA) stored in the header of an index record that is used to calculate the RBAs of data or index control intervals governed by the index record.

**basic direct access method (BDAM).** An access method used to directly retrieve or update particular blocks of a data set on a DASD.

**basic partitioned access method (BPAM).** An access method used to create program libraries on DASD for convenient storage and retrieval of programs.

**basic sequential access method (BSAM).** An access method for storing or retrieving data blocks in a continuous sequence, using either a sequential access or direct access device.

**binary coded data (BCD).** Six-bit tape characters.

**block size.** The number of records, words, or characters in a block of data; usually specified in bytes.

**buffer.** A routine or storage used to compensate for a difference in the rate of flow of data, or time of occurrence of events, when transferring data from one device to another.

**buffer pool.** A continuous area of storage divided into buffers.

## C

**channel program.** One or more channel command words that control a specific sequence of data channel operations. Execution of the specific sequence is initiated by a single start I/O (SIO) instruction.

**checkpoint.** A designated point in a program where information about a job is collected and recorded in a separate checkpoint data set for restart purposes.

**configuration.** (1) The arrangement of a computer system as defined by the characteristics of its functional units. (2) See *SMS configuration*.

## D

**data extent block (DEB).** A control block that describes the physical attributes of the data set.

**Data Facility Product (DFP).** An IBM licensed program used to manage programs, devices, and data in an MVS operating environment.

**data integrity.** Preservation of data or programs for their intended purpose. As used in this publication, the safety of data from inadvertent destruction or alteration.

**data management.** The task of systematically identifying, organizing, storing, and cataloging data in an operating system.

**data security.** Prevention of access to or use of data or programs without authorization. As used in this publication, the safety of data from unauthorized use, theft, or purposeful destruction.

**data set.** The major unit of data storage and retrieval in the operating system, consisting of data in a prescribed arrangement and described by control information to which the system has access. As used in this publication, a collection of fixed- or variable-length records in auxiliary storage, arranged by VSAM in key sequence or entry sequence. See also *key-sequenced data set* and *entry-sequenced data set.*

**data set control block (DSCB).** A control block in the VTOC that describes data set characteristics.

**data set labels.** Labels that precede and follow a data set on tape.

**density.** The amount of data that can be stored on a defined physical space. On tape density is measured in bits per inch (bpi).

**device management.** The task of defining input and output devices to the operating system, and then controlling the operation of these devices.

**disable all (DISALL).** Relationship that prevents a system from allocating or accessing data sets in a VIO storage group, a pool storage group, or individual volumes within a pool storage group.

**disable new (DISNEW).** Relationship that prevents a system from allocating new data sets in a VIO storage group, a pool storage group, or individual volumes within a pool storage group.

**discrete profile.** A RACF profile that contains security information about a single data set, user, or resource. Contrast with *generic profile.*

**distribution library (DLIB).** IBM-supplied partitioned data sets delivered on tape, containing single components or parts of components that the user restores to disk for subsequent inclusion in a new operating system.

## E

**end user.** A person in a data processing installation who requires the services provided by the computer system.

**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.

**esoteric name.** A name used to define a group of devices having similar hardware characteristics, such as TAPE or SYSDA. Contrast with *generic name.*

**exception.** An abnormal condition such as an I/O error encountered in processing a data set or file.

**export.** To create a backup or portable copy of a VSAM cluster, alternate index, or integrated catalog facility user catalog.

**external page storage.** The portion of auxiliary storage used to contain pages.

## F

**field.** In a record or control block, a specified area used for a particular category of data or control information.

## G

**generalized trace facility (GTF).** An optional OS/VS service program that records significant system events, such as supervisor calls and start I/O operations, for the purpose of problem determination.

**generation data group (GDG).** A collection of historically related non-VSAM data sets that are arranged in chronological order; each data set is known as a generation data set.

**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.

**generation data set (GDS).** One of the data sets in a generation data group; it is historically related to the others in the group.

**generic name**. A name assigned to a class of devices (such as 3380) that is derived from the IODEVICE statement in the MVS configuration program. Contrast with *esoteric name*.

**generic profile**. A RACF profile that contains security information about multiple data sets, users, or resources that may have similar characteristics and require a similar level of protection. Contrast with *discrete profile*.

**gigabyte**. 1,073,741,824 bytes.

**global shared resources (GSR)**. An option for sharing I/O buffers, I/O-related control blocks, and channel programs among VSAM data sets in a resource pool that serves all address spaces in the system.

**guaranteed space**. A storage class attribute indicating that SMS is to honor user-specified VOL=SER= specifications, and that SMS is to fail the request if space is not available on the specified volumes. This attribute also preallocates space on all volumes specified by the user.

## H

**header entry**. In a parameter list of GENCB, MODCB, SHOWCB, or TESTCB, the entry that identifies the type of request and control block and gives other general information about the request.

**head label**. Data set labels that precede a data set on tape.

**high-level name**. The first component of a qualified name. This name is found in a volume index of the CVOL Catalog.

## I

**import**. To restore a VSAM cluster, alternate index, or integrated catalog facility catalog from a portable data set created by the EXPORT command.

**initial program load (IPL)**. (1) The initialization procedure that causes an operating system to commence operation. (2) The process by which a configuration image is loaded into storage at the beginning of a work day or after a system malfunction.

## J

**job entry subsystem (JES)**. A system facility for spooling, job queueing, and managing input and output. The two types of job entry subsystems in MVS are JES2 and JES3.

## K

**kilobyte**. 1024 bytes.

## L

**leading tape mark bypass (LTM)**. Code signaling that if a leading tape mark is encountered on an unlabeled tape, it is bypassed.

**logical record**. (1) A record from the standpoint of its content, function, and use rather than its physical attributes; that is, one that is defined in terms of the information it contains. (2) A unit of information normally pertaining to a single subject; a logical record is that user record requested of or given to the data management function.

## M

**Mass Storage System**. The name for the entire storage system, consisting of the Mass Storage Facility and all devices that are defined to the Mass Storage Control.

**MVS/DFP**. An IBM licensed program which is the base for the Storage Management Subsystem.

**MVS/ESA**. An MVS operating system environment which supports ESA/370.

**MVS/XA**. An MVS operating system environment that supports 31-bit real and virtual storage addressing, increasing the size of addressable real and virtual storage from 16 megabytes to 2 gigabytes.

## O

**online**. Pertaining to equipment, devices, or data under the direct control of the processor.

**operating system**. Software that controls the execution of programs; an operating system may provide such services as resource allocation, scheduling, input/output control, and data management.

**OS control volume (OS CVOL)**. A volume that contains one or more indexes of the catalog.

**OS/VS2 MVS**. An MVS operating system that was the predecessor to MVS/XA.

## P

**page.** (1) A fixed-length block of instructions, data, or both, that can be transferred between real storage and external page storage. (2) To transfer instructions, data, or both between real storage and external page storage.

**password.** A unique string of characters stored in a catalog that a program, a computer operator, or a terminal user must supply to meet security requirements before a program gains access to a data set.

**performance.** For a storage subsystem, a measurement of effective data processing speed against the amount of resource that is consumed by a complex. Performance is largely determined by throughput, response time, and system availability.

**physical storage.** With respect to data, the actual space on a storage device that is to contain data.

## Q

**queued sequential access method (QSAM).** An extended version of the basic sequential access method (BSAM). Input data blocks awaiting processing or output data blocks awaiting transfer to auxiliary storage are queued on the system to minimize delays in I/O operations.

## R

**RACF always call.** A term for the procedure used by DFP to check all data sets automatically for discrete or generic RACF profiles to verify access authority.

**RACF authorization.** (1) The facility for checking a user's level of access to a resource against the user's desired level of access. (2) The result of that check.

**record.** A set of data treated as a unit.

**reflective strip.** A physical signal that marks the logical end of 7-track and 9-track tapes. The IBM 3480 Magnetic Tape Subsystem uses an internal mechanism rather than reflective tapes.

**resource.** Any facility of the computing system or operating system required by a job or task, including main storage, input/output devices, the central processing unit, data sets, and control processing systems.

**Resource Access Control Facility (RACF).** An IBM licensed program that provides access control by identifying and verifying users to the system. RACF authorizes access to DASD data sets, logs unauthorized access attempts, and logs accesses to protected data sets.

**Resource Management Facility (RMF).** A licensed program that monitors the availability and activity of an MVS system.

## S

**sequential access.** The retrieval or storage of a data record in: its entry sequence, its key sequence, or its relative record number sequence, relative to the previously retrieved or stored record. See also *addressed-sequential access* and *keyed-sequential access.*

**sequential access method (SAM).** An access method for storing or retrieving data blocks in a continuous sequence, using either a sequential access or a direct access device.

**sequential data set.** A data set whose records are organized on the basis of their successive physical positions, such as on magnetic tape. Contrast with *direct data set.*

**serialization.** In MVS, the prevention of a program from using a resource that is already being used by an interrupted program until the interrupted program is finished using the resource.

**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 CVOL Catalog for the data set name.

**storage administrator.** A person in the data processing installation who is responsible for defining, implementing, and maintaining storage management policies.

## T

**tape mark.** A mark on tape that indicates the beginning or end of a file or tape.

**task control block (TCB).** Holds control information related to a task.

**telecommunications access method (TCAM).** An access method used to transfer data between main storage and and remote or local terminals.

**time sharing option (TSO).** An optional configuration of the operating system that provides conversational time sharing from remote stations.

**trailer labels.** Data set labels that follow a data set on tape.

**translator.** A pate 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.

## U

**unit control block (UCB).** A control block in storage that describes the characteristics of a particular I/O device on the operating system.

**universal character set (UCS).** A printer feature that permits the use of a variety of character arrays. Character sets used for these printers are called UCS images.

**user labels.** Data set label groups that can include standard user labels on tape.

## V

**virtual storage.** Addressable space that appears to the user as real storage, from which instructions and data are mapped into real storage locations. The size of virtual storage is limited by the addressing scheme of the computing system and the amount of auxiliary storage available, rather than by the actual number of real storage locations.

**virtual storage access method (VSAM).** An access method for direct or sequential processing of fixed- and variable-length records on direct access storage devices. The records in a VSAM data set or file can be organized in logical sequence by a key field (key sequence), in the physical sequence in which they were written on the data set or file (entry sequence), or by relative record number.

**virtual telecommunications access method (VTAM).** A set of programs that control communication between terminals and application programs running under VSE, OS/VS1, and OS/VS2.

**virtual volume.** The data from a mass storage volume while it is located on a staging drive.

**volume.** A certain portion of data, together with its data carrier, that can be mounted on the system as a unit; for example, a tape reel or a disk pack. For DASD, a volume refers to the amount of space accessible by a single actuator.

**volume labels.** The first label on tape.

**volume serial number (VOLSER).** An identification number in a volume label that is assigned when a volume is prepared for use on the system.

# Index


## A

accessibility field
  ANSI data set 1 label 68
  ANSI standard data set label 1 96
  ANSI standard volume label 90
  ANSI VOL1 label 67
allocating
  generation data set 5
  tape data set 3
ANSI standard labels
  compared to IBM standard labels 61
  component support 119
  defined 61
  header labels 65
    *See also* data set label 1, data set label 2
  operating system support 58
  overview of processing 70
  protecting data through volume accessibility 67
  specification 1
  tape marks 66
  trailer labels 66
    *See also* data set label 1, data set label 2
  types 61
  use characteristics
    summary 58—60
    system generation requirements 58
    volume initialization 58
  version 1
    defined 57
  version 3
    defined 57
    installation exits 131
    processing on other MVS systems 61
  volume label
    creation 65, 80
    defined 65
    format 88—91
    processing 71
  volume organization 63
ASCII interchange tape
  conversion from EBCDIC 58, 133
  operating system support 58
assembler support 119
assigning
  volume serial numbers
    JCL statements 41—42
    system assignments 113
    using IEHINITT utility program 41—42
assigning volume serial numbers
  JCL statements 90
  using IEHINITT utility program 90
automatic volume recognition
  *See* AVR
AVR (automatic volume recognition)
  described 9
  7-track tapes 12

## B

backward read
  *See* RDBACK parameter
basic tape formats 1—2
binary data 11
bits per inch (densities) 10—13
blank tape
  bypass label processing 112
  IEHINITT utility
    ANSI standard labels 81
    IBM standard labels 29
block count
  ANSI standard data set label 1 96
  ANSI standard labels 75, 77
  IBM standard labels 24, 25, 48
  verification 77
block length
  ANSI standard data set label 2 100
  data set characteristics 24
  descrlibed 52
block padding 60
BLP subparameter 2, 112
buffer alignment block
  ANSI standard data set label 2 102
buffer offset
  ANSI standard data set label 2 102
bypass label processing
  component support 119
  specification 2
  using 112

## C

cataloged data sets
  ANSI standard labels 78
  described 5
  generation groups 5
  IBM standard labels 22, 26
  ISO/ANSI/FIPS standard labels 73
  no labels 108
channel status word
  *See* CSW
character code, EBCDIC 11
checking
  volume labels 115
checkpoint/restart
  *See* restart routine
CLOSE macro
  ANSI standard labels 80, 87
  function 9

CLOSE macro *(continued)*
IBM standard labels 29, 37
no labels 114
close routine
ANSI standard labels 80, 87
function 9
IBM standard labels 29, 37
no labels 114
closing an output data set
IBM standard labels 37
closing input data set
ANSI standard labels 80
IBM standard labels 29
no labels 110
closing output data set
ANSI standard labels 87
no labels 114
COBOL 119
completing data control block 4—5
component
support 119
concatenation
data sets
ANSI standard labels 68, 76, 79
described 6
IBM standard labels 25, 28
no labels 110
connected data sets
*See* concatenation, data sets
contents label 121—122
control characters
ANSI standard label 101
IBM standard labels 25, 53
conversion
codes
EBCDIC 133—136
Hollerith 133—139
ISCII/ASCII 137—139
data
ASCII to/from EBCDIC 58, 133
BCD to/from EBCDIC 11
creating
volume label
ANSI standard volume labels 80
IBM standard volume labels 29
creating volume label
nonstandard volume labels 105
creation date
ANSI standard data set label 1 95
IBM standard labels 47
CSW (channel status word) 13

## D

data class 3
data control block
*See* DCB
data conversion 11, 58, 133—140
data group index 5
data management routine 8
data recording characteristics 10—13
data set
accessibility 68
attributes 3—8
cataloged 5
characteristics 24
concatenated
ANSI standard labels 79
described 6
IBM standard labels 28
generation 5
multiple per volume 7
multiple volumes per data set 7, 26—29, 78—80, 86
name 23
opening 21
passed 7
processing methods 8
SMS-managed 8
writing header labels 35
writing user labels 35
data set header labels
ANSI standard labels 65
IBM standard labels 18—19
data set identifier 45, 117
data set label 1 (HDR1, EOV1, EOF1)
ANSI standard
contents 91—97
defined 57—66
format 91—97
processing 71, 91—97
writing 84
IBM standard
contents 43—49
defined 15—17
format 43—49
other systems 117
processing 19—38, 43—49
data set label 2 (HDR2, EOV2, EOF2)
ANSI standard
contents 97—102
defined 57—66
format 97—102
processing 71, 76, 97, 102
writing 84, 86
IBM standard
contents 49—54
defined 15—17
format 49—54
other systems 117
processing 19—38, 49—54
data set name
ANSI standard labels
checking before input processing 74
checking before output processing 69
duplicate name checking 75

data set name *(continued)*
ANSI standard labels *(continued)*
format in HDR1 label 93
IBM standard labels
checking before input processing 25
checking before output processing 34—35
format in HDR1 label 45
other systems 117
data set position
ANSI standard labeled tapes 2, 101
IBM standard labeled tapes 2, 52
data set protection
ANSI standard labels 67
described 59
multiple volumes
ANSI standard labels 79
IBM standard labels 28—29
processing
ANSI standard labels 81
IBM standard labeled tapes 21, 36, 48—49
specification 3
data set sequence number
ANSI standard labels 73, 83, 86
described 94
concatenated data sets
ANSI standard labels 79
IBM standard labels 28
default 3
IBM standard labels 22, 31, 46
multiple volumes
ANSI standard labels 79
IBM standard labels 28
no labels 108, 113
other systems 118
repositioning during Restart 37
specification 3
SYSOUT data sets 32
data set serial number
ANSI standard labels
*See* file set identifier
IBM standard labels 46
DCB (data control block)
completion 4—5
end-of-data routine
ANSI standard labels 76—80
IBM standard labels 25—29
no labels 110
exit routine 4
macro
completing data control block 4—5
DEN parameter 10—13
TRTCH parameter 12
user label (ANSI) exits 76, 85
user label (IBM) exits 25, 26, 38
DD statement
concatenated data sets 6
DCB parameter 5
DEN parameter 10—13
DD statement *(continued)*
DISP parameter 9
LABEL parameter 2
multiple data sets 7—8
multiple volumes 7
passed data sets 7
TRTCH parameter 12
VOLUME parameter 7
DDR (dynamic device reconfiguration) option 105
deferred user trailer label processing
ANSI standard labels 76
IBM standard labels 20, 36—37
DEN parameter
described 10—13
density
ANSI standard labels 101
conflicts, how resolved 10
DEN parameter codes 12
IBM standard labels 52
describing data sets
LABEL parameter 2—3
physical attributes 3
volume and unit information 3
describing labels 2—3
device type
retrieving cataloged data sets 5
DISP parameter
described 9
disposition of volumes
positioning parameters explained 9
dual density 10
dummy header label
ANSI standard labels 80, 84
IBM standard labels 29
dynamic device reconfiguration (DDR) option 105

## E

EBCDIC (extended binary coded decimal interchange code)
conversion codes 133—136
editor routines
volume label 115
module names 115
EMODVOL1 115
end of data set
ANSI standard labels 76—80
described 8
IBM standard labels 25—29
no labels 110
end of reel 13
end-of-data routine
*See* EODAD routine
end-of-volume
ANSI standard labels 76—80, 85—87
described 8
IBM standard labels 25—27, 35—37
no labels 110
special conditions 37, 87

EODAD (end-of-data) routine
  ANSI standard labels 76—80
  explained 8
  IBM standard labels 25—29
  routine, for no labels 110
EOV macro 8
EOV routine
  ANSI standard labels 76—80, 85—87
  IBM standard labels 25—27, 35—37
  no labels 110
  OPTCD=B 45, 93
EOV 1 label
  *See* data set label 1
EOV 2 label
  *See* data set label 2
EOV, volume label editor routine 115
error
  permanent I/O
    ANSI standard labels 85
    processing 35
exit routine
  open/EOV user exit for nonspecific tape mount requests 31
  open/EOV user exit for security verification 24, 28, 33, 36
  version 3 131
expiration date
  ANSI standard data set label 1 95
  ANSI standard labels 75, 84
  IBM standard labels 24, 47
  LABEL parameter 3
  on exisiting labels 24
external labels 121—122

## F

FEOV macro
  ANSI standard labels
    input tape 76
    output tape 76, 85
  IBM standard labels
    input tape 20, 25—26, 29
    output tape 20, 35
  no labels 110, 114
file section number
  ANSI standard data set label 1 94
file sequence number
  *See also* data set sequence number
  ANSI standard data set label 1 94
file set identifier
  ANSI standard data set label 1 93, 94
first record, verification of
  ANSI standard labels 72, 79
  bypass label processing 112
  IBM standard labels 21, 28
  no labels 108, 111
  nonstandard labels 105
format
  data set labels
    ANSI standard labels 91—102
    IBM standard labels 43—49, 54
  user labels
    ANSI standard labels 103—104
    IBM standard labels 55—56
FORTRAN 119
forward merge of the DCB 4

## G

GDG (generation data group)
  ANSI standard labels 94
  described 5
  IBM standard labels 46
generation data set
  allocating 5
generation numbers
  ANSI standard data set label 1 94
  described 5
  IBM standard labels 46

## H

HDR1 label
  *See* data set label 1
HDR2 label
  *See* data set label 2
header label
  ANSI identifiers 62
  data set 18, 65
  defined 1
  IBM identifiers 15
  user 18
high speed search, 3480 tape volume 32
Hollerith conversion codes 133—139

## I

IBM standard labels
  component support 119
  defined 1, 15
  operating system support 15
  overview of processing 19—21
  specification 1
  types 15
  volume layouts 15—17
IBM 3480 Magnetic Tape Subsystem 30
IBM 3480 tape processing 37, 87
IBM 3480 tape volume high speed search 32
identifying
  tape reels 121—122
IEHINITT program
  assigning volume serial numbers 41—42
  creating ANSI standard labels 68, 80
  creating IBM standard labels 29
  dummy ANSI HDR1 label 80, 84
  dummy IBM HDR1 label 29

input data set
ANSI standard labels 71—80
closing 80
IBM standard labels 21—29
no labels 107—111
opening 71
specifying 8
input/output support routines 8
*See also* open routine, close routine, EOV routine
installation exit
defaults 131
ISO/ANSI/FIPS 58
RACHECK (RACF) 59
WTOR 58, 131
International Organization for Standardization (ISO)
*See* ANSI standard labels
introduction to tape processing 1—13
ISCII interchange tape
*See* ASCII interchange tape
ISCII/ASCII control characters 53
ISCII/ASCII conversion codes 137—139
ISCII/ASCII 7-bit code 133
ISCII/ASCII 8-bit code 133
ISO (International Organization for Standardization) standard labels
*See* ANSI standard labels

## J

JFCB (job file control block)
data set sequence number
*See* positioning tapes
header label information 22—25
merging control block information 22—25
updating the 4—5
volume serial number 27, 31, 39—43
job and job step name
ANSI data set label 101
IBM data set label 53
job control statements 2—8
job file control block
*See* JFCB

## L

label definitions and organization 15—19
label formats 15
label identifier
data set labels
ANSI 93, 99
IBM 45, 51, 55
user labels
ANSI 103
IBM 55
volume label
ANSI 89
IBM 41
label number
data set labels
ANSI 93, 99
label number *(continued)*
data set labels *(continued)*
IBM 45, 51
user labels
ANSI 103—104
IBM 56
volume label
ANSI 90
IBM 41
LABEL parameter 2
label processing
ANSI standard labels 70
as used by other system components 119
IBM standard labels 19—21
label standard level
ANSI standard volume label 91
label type 2
label validation 59
label (volume) editor routines 115
labels, tape
defined 1—3
external 121—122
model 2
types 1—3
label, checkpoint security 122
label, nonstandard 118
LEAVE parameter
ANSI standard labels 73
described 9
IBM standard labels 22
no labels 109
LIKE keyword 5
linkage editor
data management, label processing 119
load point 13
LOOKAHEAD mounting 27, 35, 79, 85
LPALIB
volume verification routines 106

## M

magnetic tape
characteristics 10—13
magnetic tape labels
*See* labels, tape
member
NSLCTRLO 105
NSLEHDRI 105
NSLEHDRO 105
NSLETRLI 105
NSLETRLO 105
NSLOHDRI 105
NSLOHDRO 105
NSLRHDRI 105
merging
control block data 4—5
model data set label 2, 5

module names
nonstandard label routines 105
volume label editor routines 115
mount switch (UCBDMCT)
ANSI standard labels 72
IBM standard labels 21, 31
mounting volumes
in advance 9
moving tape data sets to DASD 8
multiple data sets
ANSI standard labels
end-of-volume conditions 87
input data set 73—74
output data set 84
volume organization 57—64
DD statements 7
IBM standard labels
end-of-volume conditions 36
input data set 22—25
output data set 31—35
restart procedure 37
volume organization 15—17
no labels 108—110
nonstandard labels 105
multiple volumes
ANSI standard labels
checking volume labels 72—79
creating volume label 81
switching volumes 78, 85
volume organization 57—64
DD statements 7
IBM standard labels
checking volume labels 21, 28—29
creating volume label 29
from other systems 117
switching volumes 26
volume organization 15—17
no labels 110—111
nonstandard labels 105
RACF protection 7

## N

named generation group 5
new volume labels
ANSI standard volume labels 86
IBM standard volume labels 29
nine-track tapes 10
NL subparameter 107, 109
nonstandard label processing routines
RACF
*See* RACF nonstandard labels
nonstandard label routines, module names 105
nonstandard labels component support
features 119
other systems 118
specification 1
NRZI mode 11
NSL subparameter 2
NSLREPOS routine 106

## O

OMODVOL1 115
OPEN macro
overriding INOUT and OUTIN parameters 3
processing for a tape data set 8—10
open routine
ANSI standard labels 71—76, 82—85
function of 8
IBM standard labels 21—25, 30—35
no labels 107—114
opening input data set
ANSI standard labels 71—76
IBM standard labels 21—25
no labels 107—110
opening output data set
ANSI standard labels, specified 82—85
caution 30
caution—empty data sets 111
IBM standard labels specified 30—35
no labels specified 111
open/EOV user exit
nonspecific tape mount requests 31
security verification 24, 28, 33, 36
open, volume label editor routine 115
output data set
ANSI standard labels 82—87
closing 37, 114
IBM standard labels 30—37
no labels 111—114
owner identification
ANSI standard volume label 91
IBM standard volume label 42

## P

PARALLEL mounting 27, 35, 79, 85
parity 10—13
passed data sets
defining attributes 7
password protection
*See also* data set protection
data set
input processing 69
output processing 69
described 34
descriged 24
PE (phase encoded) mode 11
phase encoded (PE) mode 11
PL/I 119
positioning tapes
ANSI standard labels 73—74, 83
IBM standard labels 22—23, 31—33
no labels 108—111, 113
using OPEN and CLOSE parameters 9

positioning volume to data set 73
processing
component considerations 119
HDR1 label 74
HDR2 label 76
user header labels 76
version differences 60
processing methods and routines 8
protection
*See also* data set protection
checkpoint 60
data set 59
data set accessibility 68
volume 59
volume accessibility 67

## R

RACF 1.7 3, 28, 31
RACF (resource access control facility)
ANSI standard labels
accessibility 91, 96
creating volume label 80
opening input data set 71—76
opening output data set 82—85
processing on the new volume 67, 86
protection with existing label 67
how to protect tape volumes under 3
IBM standard labels
checking the next volume 28
creating volume label 29
end-of-volume on output 36
opening input data set 21—25
opening output data set 30—35
password checking 24, 34—35
processing on new volumes 36
protection and data set name on existing label 34
writing data set header labels 35
multiple volumes protected by 7
nonstandard labels
opening output data set 111—112
protection 21
tape volumes defined 1
RDBACK parameter
ANSI standard labels 72, 73, 76, 86, 87
described 8
IBM standard labels 22, 25, 28
no labels 110
restriction with data conversion 12
read backward
described 25
record format
ANSI standard labels 99
IBM standard labels 24, 51
record length
ANSI standard data set label 2 100
reel label description 121
reflective strip
function 13
writing beyond
ANSI standard labels 87
IBM standard labels 36
relative generation number 5
REREAD parameter 9
Resource Access Control Facility
*See* RACF
restart routine
control block fields 126
from a checkpoint 114
function 9
IBM standard labels 37
no labels 114
table entry format 123
work areas 123—129
restrictions
label and versions 59
retention period
*See* expiration date
reverse merge of DCB 5
reverse read
*See* RDBACK
routine
NSLREPOS 106
processing 8
RPG (Report Program Generator) 119

## S

scratch tape
ANSI standard labels 81, 85, 90
IBM standard labels 31, 41
no labels 113
security protection
*See* data set protection
security status code
ANSI data set label 1 96
IBM data set label 1 48
security verification
open/EOV user exit 24, 33
sense byte 13
SL subparameter 15
SMS (Storage Management Subsystem)
data sets 8
sort/merge program 119
spanned records 60
standard labels
*See* ANSI or IBM standard labels
standard volume label
*See* volume label
SUL subparameter 15
system code
ANSI standard data set label 1 97
IBM standard labels 49

system generation
bypass label processing 112
7-track density 11—13
system generation requirements 58
system input tapes (SYSIN) 12
system output tapes (SYSOUT) 12, 32

## T

tape characteristics 10—13
tape data set
allocating 3
tape disposition 9
tape format 2
tape labels
*See* labels, tape
tape marks
ANSI standard labels 66
defined 13
how processed 13
IBM standard labels 19
no labels 108—110, 112
tape processing
IBM 3480 37
tape recording technique 13, 53, 101
tape reposition routine 105
tape units 10, 13
tapes created by other systems 117
tape, unlabeled 118
track density 10—13
trailer labels
ANSI standard
deferred processing 76
defined 66
format 91—102
data set definition 18
IBM standard
deferred processing 20
defined 1
format 43, 49
nonstandard 11—13
user 18
user label (IBM) exits 36
writing data set 36
writing user 36
translation
ASCII to/from EBCDIC 58, 133
BCD to/from EBCDIC 11
TRTCH parameter 12

## U

UCBDMCT
*See* mount switch
unit exception bit, when set 13
unlabeled tapes
component support 119
processing 107—114, 118
replaced by labeled tape 112
unlabeled tapes *(continued)*
specification 1
user identification
*See* owner identification
volume layouts 109
user labels
ANSI standard
format 103—104
location on volume 57—64
processing 76, 77
writing 85, 86
IBM standard
component support 119
deferred processing 20
format 55—56
location on volume 15—17
processing requirements 18
writing 35
utility programs 119
*See also* IEHINITT program

## V

variable-length record 12—13
verification of volume labels 115
version number of generation
ANSI standard data set label 1 95
version numbers
described 5
IBM standard labels 47
Version 1
*See* ANSI standard labels
Version 3
*See* ANSI standard labels
volume accessibility 67
volume disposition 9
volume identifier
ANSI standard volume label 90
volume initialization 58
volume label
ANSI standard
creation 65, 80
defined 65
format 88—91
processing 71, 82
editor routines 115
module names 115
IBM standard
creation 29
defined 18
format 39—42
processing 19, 21, 28—29, 30, 35
volume label verification 115
volume organization
ANSI standard labels 61—64
basic layouts 2
IBM standard labels 15—17
unlabeled tapes 107—110

VOLUME parameter 7
volume sequence number
  ANSI standard labels
    volume switching 78
  IBM standard labels
    location in header label 46
    volume switching 26
volume serial number
  ANSI standard labels
    described 90
    input processing 72
    output processing 83
    switching volumes 78
  assigned
    in JCL statements 41—42
    through IEHINITT 41—42
  assigning
    by system 113
  external labels 121—122
  IBM standard labels
    described 21
    input processing 21
    output processing 31
    switching volumes 26
  no labels 110—113
volume switching
  ANSI standard labels 78, 85
  EOV output 35
  IBM standard labels 26—27, 35
  no labels 110—111
  provisions in EOV routines 8
volumes created by other systems 117
VOL1 label
  *See* volume label

## W

work area
  restart routines 123—129
writing
  data set header labels 35
  data set trailer labels 36, 37
  header labels 35
  user header labels 35
  user labels 35
  user trailer labels 36, 37
WTOR installation exit
  convertion
    non-Version 3 to Version 3 labels 58
  defaults 131

## Numerics

18-track tapes 13
3430 Magnetic Tape Subsystem
  tape characteristics 10
3480 Magnetic Tape Subsystem
  and the open and EOV modules 30
  specifying density 13
3480 Magnetic Tape Subsystem *(continued)*
  18-track feature 13
3480 tape processing 37, 87
3480 tape volume high speed search 32
7-track feature
  code translation 11—13
  data conversion feature 11
  density requirements 12
  lack of ANSI support 11
  models allowing 11
9-track tapes 10

MVS/ESA
Magnetic Tape Labels and
File Structure Administration

SC26-4511-1

# Reader's Comment Form

This manual is part of a library that serves as a reference source for system analysts, programmers, and operators of IBM systems. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you. Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate.

**Note: Do not use this form to request IBM publications. If you do, your order will be delayed because publications are not stocked at the address printed on the reverse side. Instead, you should direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.**

If you have applied any technical newsletters (TNLs) to this book, please list them here: _____

**Comments** (please include specific chapter and page references) :

Note: Staples can cause problems with automatic mail-sorting equipment.
Please use pressure-sensitive or other gummed tape to seal this form.

If you want a reply, please complete the following information:

Name _____ Date _____

Company _____ Phone No. (_____) _____

Address _____

_____

Thank you for your cooperation. No postage is necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail them directly to the address in the Edition Notice on the back of the title page.)

## Reader's Comment Form

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, NY

POSTAGE WILL BE PAID BY ADDRESSEE

**International Business Machines Corporation**
**Department J57**
**P.O. Box 49023**
**San Jose, CA 95161-9945**

Fold and tape | Please do not staple | Fold and tape

IBM®

IBM®

Program Number
5665-XA3

File Number
S370-34

SC26-4511-1

Printed in U.S.A.