This document describes utilities at EI&O for using magnetic tapes. The utilities include CARDTAPE, DISKTAPE, TOCMS, TPUTIL, TAPEDISK.

Other types of utilities are described in other EI&O documents. For more information about tapes, see also Using Magnetic Tapes at EI&O [http://docweb.cns.ufl.edu/docs/d0017/] (D0017).

Related topics include reel and cartridge tapes, TAPE8, TAPE9, TAPEC, JCL, OS/390, batch, services.

UF Information Technology
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>CARDLIST</td>
<td>1</td>
</tr>
<tr>
<td>CARDLIST</td>
<td>1</td>
</tr>
<tr>
<td>CARDTAPE</td>
<td>3</td>
</tr>
<tr>
<td>JCL</td>
<td>3</td>
</tr>
<tr>
<td>Examples</td>
<td>4</td>
</tr>
<tr>
<td>DISKTAPE</td>
<td>7</td>
</tr>
<tr>
<td>JCL</td>
<td>7</td>
</tr>
<tr>
<td>Examples</td>
<td>8</td>
</tr>
<tr>
<td>TPUTIL</td>
<td>9</td>
</tr>
<tr>
<td>TPUTIL</td>
<td>9</td>
</tr>
<tr>
<td>Tape Compaction Support</td>
<td>9</td>
</tr>
<tr>
<td>The TPUTIL Cataloged Procedure</td>
<td>10</td>
</tr>
<tr>
<td>The OPTIONS Command</td>
<td>12</td>
</tr>
<tr>
<td>The BACKUP Command</td>
<td>14</td>
</tr>
<tr>
<td>The COPY Command</td>
<td>16</td>
</tr>
<tr>
<td>The DUMP Command</td>
<td>19</td>
</tr>
<tr>
<td>The MAP Command</td>
<td>25</td>
</tr>
<tr>
<td>The TRANSLATE Command</td>
<td>27</td>
</tr>
<tr>
<td>More TPUTIL Examples</td>
<td>29</td>
</tr>
<tr>
<td>Mapping an ASCII or Non-Labeled Tape</td>
<td>29</td>
</tr>
<tr>
<td>TAPEDISK</td>
<td>31</td>
</tr>
<tr>
<td>TAPEDISK</td>
<td>31</td>
</tr>
<tr>
<td>JCL</td>
<td>31</td>
</tr>
<tr>
<td>Examples</td>
<td>32</td>
</tr>
</tbody>
</table>
List of Figures

1. JCL to Copy Card Images to a Standard Labeled Tape ............................................3
2. Example to Copy Multiple Sets of Card Images to Tape ..........................................4
1. JCL to Copy a Disk Data Set to Tape .....................................................................7
1. Sample Job Setup to Dump a Cartridge Tape Using TPUTIL .................................12
2. Using the **BACKUP** Command to Duplicate a Cartridge Tape. ...............................16
3. Using the **COPY** Command to Copy Multiple Volumes. ..................................19
4. Providing Information for the **COPY** Command via **DD** Statements. ..............19
5. Concatenating Three Files with the **COPY** Command. .....................................19
6. Dumping Selected Records with the **DUMP** Command. ..................................23
7. Using the MAP Command for a Standard Labeled Tape. .....................................25
8. Using the **TRANSLATE** Command for an ASCII Tape. ...................................28
9. Using the **BACKUP** Command to Duplicate a Cartridge Tape. ..........................29
10. Running a Syntax Check on **TPUTIL** Control Statements. ..............................29
11. Mapping an ASCII or Non-labeled Tape ..................................................................30
1. JCL for Using the TAPEDISK Procedure .............................................................31
2. Example to Copy a File from a 9-Track Tape to Disk. ............................................33
CARDLIST

This utility allows you to copy card (80-byte records) images to a disk data set or to a tape file. For information about using this utility, see the EI&O document, *Utilities for OS/390 Disk Data Sets* (D0071) [http://docweb.cns.ufl.edu/docs/d0071/d0071.html].
CARDTAPE

The CARDTAPE procedure creates a tape data set from card images, or adds card images to the end of an existing tape data set. The required parameters are the output tape volume name (VOL) and, if it is a standard labeled tape, the tape data set name (TDSN). If you do not specify any other parameters, the output tape will be a standard labeled cartridge (TAPEC) tape with the new data set occupying the first file on the tape. (If your tape already has files on it, you must specify the FILE= parameter.) The records will have a fixed-block format with a logical record length of 80 characters and a blocksize of 6160.

JCL

The JCL required to use CARDTAPE is as follows:

Figure 1. JCL to Copy Card Images to a Standard Labeled Tape

```
//jobname JOB (,,time,lines),'your name',CLASS=class
/*ROUTE PRINT node.location
/*SETUP TAPEC,1
 // EXEC CARDTAPE,VOL=volumename,TDSN='tape.dataset.name'
 //GO.CARDS DD * ...card images...
```

Parameters

The following optional parameters can be used with the CARDTAPE procedure. Each has a default value if you do not specify the parameter. You can override the default values by specifying any of the keywords below on the EXEC statement. Each parameter must be separated by a comma.

- **TAPE=drive** specifies a cartridge tape drive (, TAPE=TAPEC) at 38000 bpi, a 9-track tape drive at 6250 or 1600 bpi (, TAPE=TAPE9), or a 9-track tape drive with a density of 1600 or 800 bpi (, TAPE=TAPE8). Default is TAPEC.

- **LABEL=type** specifies other than a standard labeled tape. Default is SL.

- **FILE=file#** specifies the file for the new data set on the output tape. Default is 1. If your tape has multiple files, you must specify the ,FILE= parameter to avoid writing over existing files.

- **TLRECL=length** specifies a new logical record length. Default is 80.

- **TBLKS=size** specifies a new blocksize. Default is 6160.
Examples

TDSN=name specifies the tape data set name (required for a standard labeled tape). Default is null.

RETAIN=RETAIN specifies that the tape should not be dismounted at the end of this job step. Default is null.

TRECFM=format specifies a new record format. Default is FB.

N1DISP=disp specifies disposition of the tape data set at the start of the job step. Default is NEW.

N2DISP=disp specifies the disposition of the tape data set at the end of the job step. Default is KEEP.

DEN=n overrides the default tape densities. The following chart should be used to determine the value of n:

<table>
<thead>
<tr>
<th>TAPEC</th>
<th>DEFAULT</th>
<th>TO SPECIFY</th>
<th>CODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAPE9</td>
<td>38000 BPI</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>TAPE8</td>
<td>6250 BPI</td>
<td>1600 BPI</td>
<td>,DEN=3</td>
</tr>
<tr>
<td></td>
<td>1600 BPI</td>
<td>800 BPI</td>
<td>,DEN=2</td>
</tr>
</tbody>
</table>

Examples

The following EXEC statement copies card images onto a tape called ARTGO1 using default parameters:

// EXEC CARDTAPE,VOL=ARTGO1,TDSN='UF.userid.JULY'

To specify a blocksize of 12000, code the EXEC statement as follows

// EXEC CARDTAPE,VOL=ARTGO1,TBLKS=12000,TDSN='UF.userid.JULY'

Figure 2 shows a job that copies several different sets of card images, placing each set in a different file on the tape. Use of the RETAIN parameter on each job step except the last will keep you from incurring multiple mount charges for the same tape.

Figure 2. Example to Copy Multiple Sets of Card Images to Tape

//jobname JOB (,,time,lines),"your name",CLASS=class
/*ROUTE PRINT node.location /*SETUP TAPEC,1 /*STEP1 EXEC CARDTAPE,VOL=ARTGO1, // FILE=1,TDSN='UF.userid.BLUE',RETAIN=RETAIN //GO.CARDS DD * ...card images for file 1... /** /*STEP2 EXEC CARDTAPE,VOL=ARTGO1, // FILE=2,TDSN='UF.userid.RED',RETAIN=RETAIN //GO.CARDS DD * ...card images for file 2...
To add card images to the end of an existing tape data set, code the EXEC statement as follows:

```c
// EXEC CARDTAPE, VOL=ARTG01,
// TDSN='UF.userid.GREEN', N1DISP=MOD, FILE=4
```
DISKTAPE

The DISKTAPE procedure copies a disk data set to an output tape or appends a disk data set to the end of an existing tape data set. If the tape data set is created successfully, it also makes a specified disposition of the original disk data set. The required parameters are the disk data set name (DDSN) and the output tape volume name (VOL). Using default values for other parameters will result in the disk data set being transferred as the first file on a cartridge tape with the same record format, logical record length, blocksize, and data set name as the disk file. The disk file would be kept.

JCL

To access DISKTAPE, use the following JCL:

Figure 1. JCL to Copy a Disk Data Set to Tape

```
//jobname JOB (,,time,lines),'your name',CLASS=class
/*ROUTE PRINT node.location
/*SETUP TAPEC,1
// EXEC DISKTAPE,DDSN='name',VOL=volume
```

DDSN= is the disk data set name.

VOL= is the tape volume name.

Parameters

You can override the default values by specifying any of the optional parameters and values below on the EXEC statement. Each parameter must be separated by a comma.

- **TLRECL=length** specifies the logical record length of the tape data set. Default is disk data set record length.
- **TRECFM=format** specifies record format. Default is disk data set format.
- **TBLKS=size** specifies a tape blocksize. Default is disk data set blocksize.
- **FILE=file#** specifies the relative file position of the new tape file. Default is 1.
- **TAPE=drive** specifies a cartridge tape drive (TAPE=TAPEC) at 38000 bpi, a 9-track tape drive at 6250 or 1600 bpi (TAPE=TAPE9), or a 9-track tape drive with a density of 1600 or 800 bpi (TAPE=TAPE8).
**Examples**

Default is **TAPEC**.

- **DISP** specifies the disposition of the disk data set upon successful job completion. Default is **KEEP**.

- **TDSN** specifies the tape file name, if different from **DDSN**. Default is the value of **DDSN**.

- **LABEL** specifies the type of tape label (**NL** or **SL**). Default is **SL**.

- **RETAIN** specifies that the tape will not be dismounted at the end of the job step. Default is null.

- **N1DISP** specifies disposition of the tape data set at the start of the job step. Default is **NEW**.

- **N2DISP** specifies disposition of the tape data set at end of the job step. Default is **KEEP**.

- **UNIT** specifies the type of disk unit. Default is null.

- **DEN** overrides the default tape densities. The following chart should be used to determine the value of \( n \):

<table>
<thead>
<tr>
<th>Tape Type</th>
<th>Default</th>
<th>To Specify</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAPEC</td>
<td>38000 BPI</td>
<td>----</td>
<td>----</td>
</tr>
<tr>
<td>TAPE9</td>
<td>6250 BPI</td>
<td>1600 BPI</td>
<td>( \text{DEN}=3 )</td>
</tr>
<tr>
<td>TAPE8</td>
<td>1600 BPI</td>
<td>800 BPI</td>
<td>( \text{DEN}=2 )</td>
</tr>
</tbody>
</table>

**Examples**

The following JCL creates a tape data set on file 1 of tape volume TAPE01 from a cataloged sequential disk data set named 'UF.GOGATOR.MYFILE' and deletes the disk data set if the creation is successful.

```
// EXEC DISKTAPE,DDSN='UF.GOGATOR.MYFILE',VOL=TAPE01,
// DISP=DELETE
```

The following JCL appends a disk data set named UF.GOGATOR.MYFILE to the end of a tape data set named UF.GOTEAM.YOURFILE and then catalogs UF.GOTEAM.YOURFILE.

```
// EXEC DISKTAPE,DDSN='UF.GOGATOR.MYFILE',VOL=TAPE01,
// TDSN='UF.GOTEAM.YOURFILE',N1DISP=MOD,N2DISP=CATLG
```
TPUTIL

**TPUTIL** is a comprehensive utility for the manipulation of magnetic tapes. It allows you to BACKUP, COPY, DUMP, MAP, and TRANSLATE a tape. In addition, it includes support for the use of tape compaction with 3480 cartridge tapes. When you request the same tape in multiple commands, the tape is retained on the tape drive (when possible) to save the cost of a tape mount for each command. The following summarizes the **TPUTIL** commands.

**BACKUP**

Makes a backup copy of all or part of a tape volume onto a second tape.

**COPY**

Copies a tape volume, individual files, or a multi-volume tape data set. The **COPY** command allows you to change many tape data set characteristics during the copy operation, while **BACKUP** makes an exact duplicate of your tape.

**DUMP**

Allows you to dump all or part of one or more tape files in character and/or hexadecimal format.

**MAP**

Requests a report of the contents of a tape. For standard labeled tapes, this produces information from the labels. For non-labeled tapes, it lists the first 80 bytes of each file.

**TRANSLATE**

Modifies execution of the **DUMP** command to convert the data from other character codes, such as ASCII.

**OPTIONS**

Allows you to specify options that will be used throughout the execution of your **TPUTIL** job.

**Tape Compaction Support**

All **TPUTIL** commands that print tape characteristic information (e.g., **MAP**) will indicate if a 3480 cartridge is compacted. This information is derived from the tape labels or parameters on the **TPUTIL** commands.

Output volume label processing in the **BACKUP** and **COPY** commands will add the **TRTCH** value to the label and always rewrite the volume label when file #1 is written. This handles compacted cartridges correctly and allows the density on tape reels to be changed by the **BACKUP** command.

Whatever **TRTCH** value is written to the volume label defines the **TRTCH** value for the entire tape. That is, if file #1 is written using compaction, all other files written to that tape must also be written using compaction. You cannot mix files with different recording techniques on the same tape.

Also, suppose that you have a cartridge tape written before tape compaction became
the default. Now, you copy it to another cartridge, beginning with file #1, but do not specify a
TRTCH value. The TRTCH value of your output tape will be COMP, the system default. If
this is not what you wish, you must specify TRTCH=NOCOMP as described in the commands
below.

You might wonder what will happen if you copy a TRTCH=COMP cartridge tape to a 9-track
tape without specifying TRTCH=NOCOMP. Your job will run successfully. The system
recognizes the type of tape you are using and will act accordingly.

The **TPUTIL** Cataloged Procedure

**Note**

Throughout this chapter, optional parameters are shown in angle brackets (< >).

The **EXEC** statement for **TPUTIL** has the following format. All of the parameters are
optional. If you do not use any parameters, one TAPEC tape drive is assumed.

```
// EXEC TPUTIL,<,PARM=parm>,
// <,DSN2=dsn>,
// <,UNIT2=unit>,
// <,VOL2=volume>
```

The parameters have the following meanings.

**PARM**

specifies an input control statement to **TPUTIL**, allowing you to simplify the coding of some runs, where "parm" is a complete **TPUTIL** control statement. (See the example below.)

**DSN1, DSN2**

are used for the allocation of the appropriate number of tape units for any particular step. The default is DSN1= and DSN2=NULLFILE. This allocates one tape drive (as for a **MAP**). If you only need one tape drive, do not use DSN1 or DSN2.

If you need two tape drives (e.g., for a **BACKUP** or **COPY**), specify DSN2= on the **EXEC** statement, as follows:

```
/*SETUP TAPEC,2 // EXEC TPUTIL,DSN2=
```

or

```
/*SETUP TAPE9,1
/*SETUP TAPEC,1
// EXEC TPUTIL,UNIT1=TAPE9,DSN2=
```

or

```
/*SETUP TAPE9,2
// EXEC TPUTIL,UNIT1=TAPE9,UNIT2=TAPE9,DSN2=
```

If you do not need any tape drives (as for an edit run), code **DSN1=NULLFILE** as follows:
EXEC TPUTIL,DSN1=NULLFILE

Note

Do not forget to add or delete the appropriate JCL SETUP statements.

UNIT1 is the type of tape drive to be allocated for the input volume. The default for "unit" is TAPEC.

UNIT2 is the type of tape drive to be allocated for the output volume, if any. The default for "unit" is TAPEC.

VOL1, VOL2 are the volume serial of a tape retained in a previous step to be passed to this step, if needed.

Command Syntax

A TPUTIL control statement is a command (e.g., BACKUP) with all necessary parameters. All control statements and the parameter field of the EXEC statement are scanned by a single scan routine.

• You can use abbreviations for keywords. Minimal abbreviations for the keywords are indicated by capital letters in this chapter. For example, the minimal abbreviation for the keyword "Linesize" is "L".

• The majority of parameters are optional. If you do not specify a parameter, TPUTIL will use the default value (or derive the value from your JCL DD statements). In this chapter, any TPUTIL command parameter that is not indicated as required is optional and is enclosed in angle brackets (< >).

• When you specify multiple operands for a keyword parameter, enclose the list in parentheses. For example, to specify both the NOCHECK and NLFORCE options of the DUMP command, code the following on the control statement: OPTIONS=(NOCHECK,NLFORCE)

• The keywords can be listed in any order on the control statement.

• To continue a control statement, code a comma immediately after the last parameter on the first line; then begin the next parameter on the next line. You can use up to five continuation lines for one control statement.

• You can use the PARM= field of the EXEC statement to supply many control statements and simplify the coding of many runs. For example,
  // EXEC TPUTIL, PARM='MAP VOL=MYTAPE'

• You can use all 80 columns of the control statement unless you specified the SEQUENCE option as described in the OPTIONS command. When the SEQUENCE option is used, only columns 1-72 are scanned for control information.
Example

Figure 1 shows a simple JCL setup to dump the contents of a cartridge tape. This job will cause the tape MYTAPE to be dumped using all default parameter values.

Figure 1. Sample Job Setup to Dump a Cartridge Tape Using TPUTIL

```
//jobname JOB (,,time,lines),'your name',CLASS=class
/*ROUTE PRINT node.location
/*SETUP TAPEC,1 // EXEC
TPUTIL,PARM='DUMP VOL=MYTAPE'
```

Retaining Tapes

TPUTIL retains a tape on the tape drive if it can. In some cases, however, TPUTIL might request a second mount of the same tape. To avoid a re-mount and save a few seconds and the additional charge, use the following procedures:

1. Specify RETAIN in the prior step(s).
2. Specify VOL1=xxxxxx on the TPUTIL EXEC statement.
3. Have the first TPUTIL control statement refer to that volume.

For example, if you want to write on the tape in the first step of a job and then map or dump it using TPUTIL in the next step, specify VOL=(,RETAIN,SER=xxxxxx) on the DD statement used to write the tape in the first step. This will keep the tape on the drive. Next, specify the VOL1=xxxxxx keyword parameter on the TPUTIL EXEC statement to keep the tape from being re-mounted. If the first TPUTIL control statement refers to a volume other than the one specified with the VOL1= keyword, the volume will be dismounted and the one specified in the control statement will be mounted, even if you later reference the volume with a control statement.

The OPTIONS Command

The OPTIONS command lets you specify several options (such as length of a line or lines per page) that will be used throughout the execution of the program. If you use the OPTIONS command, it must come before the commands you want it to effect. The command itself and all of its parameters are optional.

```
OPTIONS <,PAGESIZE=n>,<,OPTIONS=options>, <,DELIMITER=string>
```

Linesize=n specifies the maximum number of characters per line (70-132) on the output print data set. A line size of 112 will give 80 characters for dumps in the VHEX and CHAR formats. The
ABEND format, MAP command and BACKUP command all require at least 120 characters per line. The default value is LINESIZE=132.

**Pagesize=n** specifies the maximum number of lines per page (15-32767) on the output data set. The default is PAGESIZE=82.

**Title=string** specifies a title ('string') that will appear in the heading of each printed page. This string can be up to 80 characters long and will be truncated as necessary to fit in the heading. The string must be in single quotes if it contains blanks or special characters.

**Delimiter=string** specifies the character string to be placed at the end of every record dumped. The string length is limited to two characters.

**Options=options** specifies any of the following execution options:

- **Go** (Default) continues processing on the next valid command when a parameter statement is found to be in error.

- **Nogo** will stop processing when a parameter statement is found to be in error. All remaining parameter statements will be listed.

- **Edit** checks the syntax of the control statements. Does not attempt to execute any commands.

- **Sequence** ignores columns 73-80 because they contain sequence numbers. The sequence operand is effective with the next input statement so if all of your job's statements are sequenced, specify **SEQUENCE** on the EXEC statement as follows:

  ```
  // EXEC TPUTIL, 
  // PARM='OPTIONS=SEQ'
  ```

- **Term** suppresses all line skips and page headings on the printed output. This is used mainly for terminal output.
The **BACKUP** Command

The **BACKUP** command makes a backup copy of a tape volume. You can backup the entire tape, which will make a duplicate copy of your original tape, or select specific files to back up. Throughout this description, operand one applies to the input tape volume and operand two applies to the output tape volume.

The **BACKUP** command requires the use of two tape drives. You must specify `DSN2=` on the EXEC statement as described above. Note: Since NERDC has only one `TAPE8` drive, it is not possible to **BACKUP** or **COPY** a `TAPE8` tape to another `TAPE8` tape.

The **VOLUMES** parameter is required; all others are optional.

**BACKUP**

```
VOLUMES=(vol1,vol2),
<SELECT=selopt>,
<DENSITY=(den1,den2)>, <FILES=(n,n,...,n)>,<OUTFILE=n>,
<ERRCNT=n>,<RETRIES=n>, <DDNAME=(ddname1,ddname2)>,
<TRTCH=(trtch1,trtch2)>, <OPTIONS=options>
```

- **Volumes=(vol1,vol2)** (Required) supplies the volume serial of the tape volumes to be used in the backup operation. The input volume is `vol1` and the output volume is `vol2`.
- **Select=selopt** specifies one of the following select options:
  - **All** (Default) selects all records and labels that do not cause an I/O error when read.
  - **Records** selects all records that do not cause an I/O error when read.
  - **Label** selects labels when the `EVEN` or `ONLY` operand is used.
  - **Even** selects all records, even if an I/O error occurs when they are read.
  - **Only** selects only the records that cause an I/O error when read.
**Density=(den1,den2)** specifies the tape density with which the tapes are recorded. The default values are the highest density supported on the tape drives used. This is ignored for cartridge tapes. For 9-track tapes, valid densities at NERDC are:

<table>
<thead>
<tr>
<th>4</th>
<th>6250 bpi</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1600 bpi</td>
</tr>
<tr>
<td>2</td>
<td>800 bpi</td>
</tr>
</tbody>
</table>

**Files=(n,...,n20)** specifies up to 20 input tape file numbers. Only these files are copied to the output tape. The default is to copy all files.

**OUtfile=n** specifies the first file to be used on the output tape. File n will be rewritten if it exists and replaced by the first file selected from the input tape. Any remaining files to be copied from the input tape will be copied to successive files on the output tape. The default is n=1.

**Errcnt=n** specifies the number of I/O errors to allow in one file on either the input or output tape. When this value is exceeded, the backup operation is aborted. The default is 10.

**Retries=n** specifies the number of retries to attempt before defining any error indication as a permanent I/O error. The default is 10.

**DDname=(ddname1,ddname2)** specifies alternate ddnames to be used in the backup operation. The default is (SYSUT1,SYSUT2).

**Trtch=(trtch1,trtch2)** specifies the type of tape compaction to be used. This applies to cartridge tapes only. The default is to use the system default, which is to compact cartridge tapes, or to use the information derived from the tape label. See the section on tape compaction support.

**Opinions=options** specifies the following options for the **BACKUP** command:

- **Check**
  - See (Default) verifies that the volume serials of both tapes are those requested on the parameter statement. This can only be done when the tapes are standard labeled.

- **NOcheck**
  - See does not verify that the correct input tape is mounted. This does not affect the output tape label verification.
The **COPY** Command

The **COPY** command copies a tape volume, a multi-volume tape data set, or individual tape files. It allows you to alter many file characteristics when the tape is copied. **COPY** will pick up many tape and data set characteristics from your JCL, tape labels, and command parameters. It is possible to alter many data set characteristics when a tape is copied using this command.

You may also use the **PARAM** keyword on the **EXEC** statement to specify the **COPY** command and provide all other information about the tapes via the **SYSUT1** and **SYSUT2** DD statements.

Throughout this description, operand one applies to the input tape volume and operand two applies to the output tape volume. Note that the **COPY** command requires the use of two tape drives, so you must specify **DSN2=** on the **EXEC** statement.

All of the parameters on the **COPY** command are optional. If you wish to use the default value of a parameter, or if you provide the information via DD statements in your JCL, you do not need to include corresponding **COPY** parameters.

```plaintext
COPY <VOLUMES=((vollst1), (vollst2))>,
<DENSITY=(den1, den2)>, <TRTCH=(trtch1, trtch2)>,
.FILES=(n,...,n20)> ,<OUTFILE=n>,
<LABELS=(lb1, lb2)>, <LRECL=(lrecl1, lrecl2)>,
<RECFM=(rf1, rf2)>, <BLKSIZE=(blk1, blk2)>,
<DSNAME=(dsn1,...,dsn10)>, <STARTAFTER=n>, <STOPAFTER=n>,
<SKIP=(n, PICK=m)>, <DDNAMES=(ddname1, ddname2)>,
<OPTIONS=options>
```

*Volumes=((vollst1), (vollst2))*

supplies the volume serial of the tape volumes to be used in the copy. The input volume list is vollst1 and the output...
volume list is vollst2. Each list may contain up to three volumes.

To copy one tape to another:

\[ \text{VOL}=((\text{intape}), (\text{outtape})) \]

To copy a multiple tape data set (e.g. one that spans two tapes)

\[ \text{VOL}=((\text{intape}), (\text{outtape})) \]

To copy a multiple tape data set (e.g. one that spans two tapes)

\[ \text{VOL}=((\text{intape1, intape2}), (\text{outtape1, outtape2})) \]

or

\[ \text{VOL}=((\text{intape, intape2}), (\text{outtape1})) \]

\[ \text{DEnsity}=(\text{den1, den2}) \]

specifies the tape density with which the tapes are recorded. The default values are the highest density supported on the tape drives used.

\[ \text{Trtch}=(\text{trtch1, trtch2}) \]

specifies the tape recording technique used. The default is TRTCH=null which results in COMP (compaction) for cartridge tapes. See the section on tape compaction support.

\[ \text{Files}=(n,...,n20) \]

specifies up to 20 input tape file numbers. Only these files are copied to the output tape. The default is to copy all files. It is not necessary that all files specified in the series to be copied have the same RECFM, LRECL, and BLKSIZE.

\[ \text{OUtfile}=n \]

specifies the first file to be used on the output tape. File n will be rewritten and replaced by the first file selected from the input tape. Any remaining files to be copied from the input tape will be copied to successive files on the output tape. The default is n=1.

\[ \text{DDnames}=(\text{ddname1, ddname2}) \]

specifies alternate ddnames to be used in the copy operation. The default is (SYSUT1, SYSUT2).

\[ \text{Recfm}=(rfl, rf2) \]

specifies the record format of the input and output files. By default the input record format is set from the input tape label and the output record format is set from the input record format.

\[ \text{LRecl}=(lrecl1, lrecl2) \]

specifies the input and output logical record length. By default the input lrecl is set from the input tape label and the output lrecl is set from the input lrecl.
**Blksize=(blk1,blk2)** specifies the input and output block size. By default the input block size is set from the input tape label and the output block size is set from the input block size.

**Labels=(lb1,lb2)** specifies the tape label characteristic of the tape volumes (SL, NL, etc.). The default is to preserve the existing label characteristics.

**DSnames=(dsn1,...,dsn10)** specifies up to 10 data set names for files on the output tape. If you want to change the data set names while copying more than 10 files, you will need to issue two COPY commands. Even though the FILES parameter allows you to specify more than 10 files at a time, the DSnames parameter will only name up to 10 at a time.

**STArtafter=n** starts to copy after skipping n records. The default is to copy all records.

**STOpafter=n** stops copying after copying n records. The default is to copy all records.

**SKip=(n,Pick=m)** copies m records, then skips n records. To copy every 10th record (beginning with and including record 1), specify SKIP=9. To copy the first 2 records out of every 10, specify SKIP=(8,PICK=2). The default is SKIP=(0,PICK=1), or copy all records.

**OPTions=OPTions=(option1,option2,...)** lists options that can be specified for the COPY command:

- **Check** (Default) verifies that the volume serials of both tapes are those requested on the parameter statement. This can only be done when the tapes are standard labeled.

- **NOCcheck** does not verify that the correct input tape is mounted. This does not affect the output tape label verification.

- **Force** ignores any end-of-volume indication.

- **Merge** creates one output file from the specified input files.

- **NLforce** treats the input tape as if it were non-labeled.

**Note**

COPY uses QSAM for all data file access and uses EXCP I/O for input tape label processing only.

**COPY Examples**

In this example, there is a tape file that spans tapes intape1 and intape2. Files on these tapes
will undergo compaction and will then be copied to tape outtape1 and outtape2 as space is needed without changing any other characteristics.

**Figure 3. Using the COPY Command to Copy Multiple Volumes.**

```
//jobname JOB (,,time,lines),'your name',CLASS=class
/*ROUTE PRINT node.location
/*SETUP TAPEC,2
// EXEC TPUTIL,DSN2= COPY VOLUMES=((intape1,intape2),
// (outtape1,outtape2))
```

This example uses the PARM keyword on the EXEC statement to invoke the COPY command. Information about the tapes is passed via DD statements TAPE1 (the input tape) and TAPE2 (the output tape).

**Figure 4. Providing Information for the COPY Command via DD Statements.**

```
//jobname JOB (,,time,lines),'your name',CLASS=class
/*ROUTE PRINT node.location
/*SETUP TAPEC,2
// EXEC TPUTIL,DSN2=,PARM='COPY DD=(TAPE1,TAPE2)'
//TAPE1 DD UNIT=(TAPEC,,DEFER),DISP=(,KEEP),VOL=SER=intape
//TAPE2 DD UNIT=(TAPEC,,DEFER),DISP=(,KEEP),VOL=SER=outtape
/*
```

This example will copy files one, three, and five from tape intape and merge them into a single file. It will then copy the merged file to file #2 on the tape outtape.

**Figure 5. Concatenating Three Files with the COPY Command.**

```
//jobname JOB (,,time,lines),'your name',CLASS=class
/*ROUTE PRINT node.location
/*SETUP TAPEC,2
// EXEC TPUTIL,DSN2= COPY
VOLUMES=((intape),(outtape)),FILES=(1,3,5),OUTFILE=2, OPTIONS=MERGE
```

The DUMP Command

The DUMP command lets you dump all or part of a tape in character and/or hexadecimal format. The DUMP command has the parameters shown below. The VOLUME parameter is required; all others are optional.

```
DUMP VOLUME=serial,<,SELECT=selopt,<,FORMAT=foropt>,
// <FILE=(n,n,...,n)>,<,SKIP=(n,PICK=m)>,<,STARTAFTER=n>,
// <STOPAFTER=n>,<,LAST=n>,<,LRECL=n>,<,RECFM=rcfm>,
// <DENSITY=den>,<,TRTCH=trtch>,<,ERRCNT=n>,<,RETRIES=n>,
// <BLKSIZE=n>,<,DDNAME=ddname>,<,OPTIONS=options>
```
**The DUMP Command**

*Volume=serial*  
(Required) provides the volume serial of the tape to be dumped.

*Select=selopt*  
determines if records and/or labels are to be formatted and dumped.

- **All**  
  (Default) selects all labels and any records that do not cause an I/O error.

- **Record**  
  selects any records that do not cause an I/O error.

- **Labels**  
  selects all labels.

- **Even**  
  selects all labels and all records (including those that cause I/O errors).

- **Only**  
  selects all labels and only those records that cause I/O errors.

Labels and records selected can be modified by the use of other keywords in the **DUMP** command (for example, **FILE**, **LAST**, **SKIP**, etc.) described below.

*FORmat=foropt*  
selects the format used in printing the dump of all selected records.

- **Char**  
  (Default) dumps any selected records in a 100-character-line format.

- **Abend**  
  dumps any selected records in an ABEND dump format.

- **Hex**  
  dumps any selected records in a hexadecimal format. The format will be 100 characters wide (2 hex digits for each of 50 bytes).

- **Vhex**  
  dumps any selected records in a vertical hexadecimal format. This format will have 100 bytes across the page with each 100 bytes taking 4 lines to print.

- **None**  
  dumps any selected records in a 132 characters-per-line format.
Refer to the `LINESIZE` parameter of the `OPTIONS` command for more information about specifying your output line length.

`File=(n,...,n20)` specifies up to 20 different files to be dumped. The default is to dump all files.

`Skip=(n,PICK=m)` dumps m records, then skips n records. To dump every 10th record (beginning with and including record 1), specify `Skip=9`. To dump the first 2 records out of every 10, specify `Skip=(8,Pick=2)`. The default is `Skip=(0,Pick=1)`, or dump all records.

`Startafter=n` starts to dump after skipping n records. The default is to dump all records.

`Stopafter=n` stops dumping after dumping n records. The default is to dump all records.

`Last=n` dumps the last n records. The default is to dump records as selected by other parameters. If the `skip`, `Startafter`, and `Stopafter` parameters are not used, and `Last` is specified, only the last n records are dumped.

`Recfm=recfm` specifies the record format used to unblock the records from the tape. The correct record format will be assumed for OS standard labeled (SL) tapes. For NL tapes, the default is `Recfm=U`; for SL tapes, the default is whatever is specified on the tape labels.

`Blksize=n` specifies the maximum blocksize used on the tape. The default is either 32,760 or the value specified in the tape labels for a standard labeled tape.

`LRecl=n` specifies the logical record length to be used in unblocking the tape. This will override the value in the tape label if used with a standard labeled tape. The default for a non-labeled tape is the full block. (Note: to override the `LRECL` value, `RECFM` will probably have to be specified also.)

`Density=den` specifies the density (den) of the tape. The default is the highest density supported on the tape drive involved. Possible parameters are 2, 3, or 4 for 800, 1600, or 6250 bpi, respectively. Density is ignored for cartridge tapes.

`Errcnt=n` allows n I/O errors to occur before cancelling the dump of this tape (default is 10). You must specify the `Errcnt` parameter if you specified `Select=EVEN` or `ONLY` and you want an error cutoff.
The **DUMP** Command

**RETryes=n**
retries n times before declaring an I/O error (default is 10).

**DDname=ddname**
specifies the ddname to be used by the **DUMP** command (default is **DDNAME=SYSUT1**).

**Options=options**
specifies many different options of the **DUMP** command.

- **CHeck**
  (Default) checks the input tape's labels (if any) to verify that the correct tape is mounted

- **NOCheck**
does not check the input tape's labels to verify that the correct tape is mounted

- **Punch**
  allows you to dump records from your tape to another tape or disk data set. Use **DDname** **SYSPUNCH** when identifying the output data set and **DCB** characteristics in your JCL. The **LRECL** is not limited to 80 bytes.

- **NOPrint**
suppresses printing of the dumped records. You may wish to use this option when using **PUNCH** to dump records to a data set.

- **Force**
  ignores any end-of-volume indication.

- **NLForce**
treats any tape as being non-labeled

- **COut**
counts the number of records and blocks in each file. This parameter will increase the amount of I/O if the entire file is not being dumped. It can therefore increase the charges for the job.

- **Block**
does not block any of the selected blocks (treats all blocks as if they were **RECFM=U**)

**DUMP** Example

---

22 Utilities for Magnetic Tapes
This job will dump 30 records from files 3 and 10, beginning with record number 13 of each file.

Figure 6. Dumping Selected Records with the **DUMP** Command.

```plaintext
// Jobname JOB (,,time,lines),'your name',CLASS=class
/*ROUTE PRINT node.location
/*SETUP TAPEC,1
// EXEC TPUTIL DUMP VOL=MYDATA,START=12,STOP=30, FILE=(3,10)
```

**Interpreting DUMP Command Output**

Output from the **DUMP** command provides information in four major components. They are the Volume Label, the file Header Label, the file data, and the file Trailer Label. The latter three components are repeated as many times as needed to represent all files on the tape. The **DUMP** command helps you see the contents of tape files. Sometimes, it can also be used to "map" a tape when the **MAP** option cannot be used. The following is a brief explanation of each output item.

1. **Volume Label.** This label identifies basic tape volume attributes.
   - **INPUT COMMAND STATEMENTS.** This is a list of user options that you specified for this job.
   - **VOLUME SERIAL SPECIFIED.** This shows the name of the tape volume that you specified.
   - **ACTUAL TAPE DENSITY.** This is the density in bits per inch (BPI) of the volume mounted on the tape drive.
   - **TAPE IS **L**ABELED.** This indicates the type of label found on this tape. Usually, it is standard-labeled (**SL**) or non-labeled (**NL**).
   - **VOLUME SERIAL.** This shows the internal-volume name found on the tape that is currently mounted. Compare this with VOLUME SERIAL SPECIFIED. If they do not match, then the job will be cancelled. You can specify **OPTION=NLFORCE** to continue dump processing.
   - **TAPE RECORDING TECHNIQUE.** This tells you if your cartridge tape was recorded in compacted format or non-compacted format. This is especially important if you plan to take this tape to another site. Not all 3480 cartridge drives read/write in compacted format.

2. **Header Label (HDR1).** The Header Label gives basic information about a particular file on the tape.
   - **DATA SET NAME.** The data set name is the name of the file on the tape. Notice that only the last seventeen (17) characters of the name are recorded. When you bring a tape file back to disk, you will need to specify the entire fully-qualified data set name, not just the last seventeen characters.
- **DATA SET SERIAL NUMBER.** This is another place that your tape name (volume/serial) is listed.

- **VOLUME SEQUENCE NO.** This tells you which tape number this is in a sequence of tapes. For example, if you have two volumes of data, this field would specify which tape was mounted (1 or 2).

- **DATA SET SEQUENCE # (FILE #).** This specifies the relative position of a file on the tape. It also lets you see how many files are on a tape which is important if you wish to add a file without overwriting existing files.

- **CREATION DATE.** This is the date the file was written to tape.

- **EXPIRATION DATE.** This field is used in production work to determine when a tape may be recycled. It does not mean that the data will be lost or changed on this date.

- **DATA SET SECURITY.** This indicates whether or not a password is required to access this data set.

3. **Header Label (HDR2).** The HDR2 label provides additional information about a file on the tape.

- **RECORD FORMAT.** The record format gives you information about the records in a file. For example, are they all the same length? A more complete explanation of record formats can be found in any system 370/390 JCL manual.

- **BLOCK LENGTH.** This field specifies the size of a block of data. BLOCK LENGTH must always be a multiple of the RECORD LENGTH. So, to determine the number of records per block, divide the BLOCK LENGTH (also called Block Size) by the RECORD LENGTH.

- **RECORD LENGTH.** This tells you how many bytes of data are in each record.

- **TAPE DENSITY.** This specifies the number of bits per inch (BPI) recorded on the tape. This may be important if you plan to take this tape to another site.

- **DATA SET POSITION.** This field tells you if more than one tape volume was needed to read a particular file. A zero means no; a one means yes.

- **CREATION JOB/JOB STEP.** This represents the name of the job, and the job step name, that was used to create this data set.

- **TAPE RECORDING TECHNIQUE.** This will be either **COMPACTED** or **NOCOMP** (not compacted). Only cartridge files may be compacted. If you plan to take your tape to another site, you should copy it in **NOCOMP** format first.

- **PRINTER CONTROL CHARACTER.** This indicates whether or not printer carriage control characters appear in column one of the file.

- **BLOCK ATTRIBUTE.** This describes how the data is arranged on the tape. Normally, it will be blocked data.

4. **Data.** The number of records you specify will be printed. If your tape is non-labeled, the
system will print blocks of data instead of records.

5. Trailer Label. The trailer label specifies the same information as the Header Label, with the addition of the "BLOCK COUNT" field that tells how many blocks are used to store file records.

The **MAP** Command

The **MAP** command requests a compressed listing of the tape labels for a standard labeled tape. The listing includes the file number, data set name, record format, block length, block count, density, tape recording technique, creation date, and expiration date.

For *NL* tapes, the **MAP** command lists the first 80 bytes of each file.

The **VOLUME** parameter is required; the others are optional.

\[
\text{MAP VOLUME=} \text{volser} <, \text{DDNAME=} \text{ddname} <, \text{OPTIONS=} \text{options} >
\]

- **Volume=volser** (Required) provides the volume serial of the tape to be mapped.
- **Ddname=ddname** specifies the ddname to be used for the **MAP** operation. The default is *DDNAME=SYSUT1*.
- **Options=Nocheck** tells the **MAP** processor not to verify that the correct standard labeled tape is mounted.

**MAP Example**

This job produces a file map of the standard labeled tape STRTRK.

**Figure 7. Using the MAP Command for a Standard Labeled Tape.**

\[
//\text{jobname JOB (,,time,lines),'your name',CLASS=class} \\
/*\text{ROUTE PRINT node.location} \\
/*\text{SETUP TAPEC,1} \\
//\text{EXEC TPUTIL MAP VOL=STRTRK}
\]

**Interpreting MAP Command Output**

Your output from the MAP command provides ten items of information about the data stored on the tape. This information will be in a table on the page titled, 'FILE MAP OF VOLUME xxxx', where xxxx is the name of your tape. The following is a brief explanation of each output item.

**FILE NUMBER**

This specifies the position of a file on the tape. It also lets you see how many files are on a tape. This is especially important if you wish to add a file to a tape without overwriting existing files.
DATA SET NAME
This specifies the name of a file on the tape. Notice that only the last seventeen (17) characters of the name are recorded. When you bring a tape file to disk, you will need to specify the entire fully-qualified data set name.

RECORD FORMAT
The format of a file may be specified by one or more of the following characters. Most tape files are FB. This means that the record lengths are all the same and that they are recorded in blocks. A more complete explanation of these types can be found in any system 370/390 JCL manual.

- First Character

  F  Fixed-length records
  V  Variable-length records
  U  Undefined-length records
  D  Variable-length ASCII tape records

- Next Characters (can be more than one)

  B  Blocked records
  S  Spanned records (spanned over more than one block)
  A  An ASA control character is the first byte of each record to control the printer.
  M  A machine code control character is the first byte of each record and is used to control the printer.

BLOCK LENGTH
This field specifies the size of a block of data. BLOCK LENGTH must always be a multiple of the RECORD LENGTH. So, to determine the number of records per block, divide the BLOCK LENGTH (also called Block Size) by the RECORD LENGTH.

RECORD LENGTH
This tells you how many bytes of data are in each record.

BLOCK COUNT
This reports the number of blocks of data each file contains. You can use the BLOCK COUNT to determine the number of records in a file by multiplying the BLOCK COUNT by
The number of records per block (see BLOCK LENGTH).

**DENSITY**

This specifies the number of bits per inch (BPI) recorded on the tape. This may be important if you plan to take this tape to another site.

**TAPE RECORDING TECHNIQUE**

This will be either COMP (compacted) or NOCOMP (not compacted). Only cartridge files may be compacted. If you plan to take your tape to another site, you should copy it in NOCOMP format first.

**CREATION DATE**

This is the date this file was written to tape.

**EXPIRATION DATE**

This field is used in production work to determine when a tape may be recycled. It does not mean that the data will be lost or changed on this date.

---

**The TRANSLATE Command**

The **TRANSLATE** command modifies the execution of the **DUMP** command to make data recorded in other character codes easier to understand. **TPUTIL** provides some of the common translate tables. You can also code your own translate table if necessary.

The command itself and all of its parameters are optional. If you use the **TRANSLATE** command with no parameters, the **DUMP** command will assume standard EBCDIC character code (the same as no **TRANSLATE** command).

**TRANSLATE <TO=trtb1><,FROM=trtb1>,
// <type,(string1,string2)><, START=hh><,FILL=string>**

---

**TPUTIL-Defined Translate Tables**

**TPUTIL** has several pre-defined translate tables. To access one of these tables, use either the **TO** (translate TO a specific character set on output) or **FROM** (translate FROM a specific character set on input) keyword. The translate tables that can be specified are the same for each keyword. Both keywords can be coded if needed.

**TPUTIL** provides the following translate tables (minimal abbreviations are shown in upper case):

- **EBCDIC**
  (Default) IBM System/370 standard character code

- **ASCII**
  ASCII 7-bit code with no parity

- **ASCII-Even**
  ASCII 7-bit code with even parity

- **ASCII-Odd**
  ASCII 7-bit code with odd parity

- **BCD**
  IBM 6-bit code used on the 1401. This code is also used by many other types of six-bit machines.

- **CDC-console**
  Control Data Corporation console code. This is used for some source tapes sent from CDC installations.
User-Defined Translate Tables

You only need the remaining parameters when one of the translate tables above does not apply. The discussion of these parameters assumes that you are familiar with the concepts of the "TRANSLATE" machine instruction and its associated translate table.

**type** specifies the input format of data to be placed in the translate table.

- **Character** (Default) specifies that each translate table location will be represented by a single character on the input statement.
- **Hex** specifies that each translate table location will be represented by the two-digit hexadecimal representation of the character.

If only the first string is specified, it is placed in the next empty location in the translate table or the location specified by the **START** parameter. When both string1 and string2 are specified, string1 is used as a location string and string2 is used as a replacement string. In other words, the character at offset string1 in the translate table is replaced with string2. (Note that this is a character-to-character replacement.)

**FILL=string** The **FILL** parameter can be specified only on the first **TRANSLATE** control statement following the **DUMP** command. The string is either a single character or the hex representation of the character to be used to fill all locations in the translate table before placing any other strings in the table. The **TYPE** operand indicates which input format to expect.

**Start=hh** specifies the first location in the translate table to place string1. The value **hh** is always in hex.

**TRANSLATE Example**

This job will read an ASCII tape. Each block will be broken into 80-character records. Each record will then be translated from the ASCII character to the EBCDIC character and printed. The title specified in the **TITLE** option of the **OPTION** statement will be printed on each page of output.

**Figure 8. Using the **TRANSLATE** Command for an ASCII Tape.**

// jobname JOB (,,time,lines),'your name',CLASS=class
//*ROUTE PRINT node.location
//*SETUP TAPE9,1
// EXEC TPUTIL,
// UNIT1=TAPE9 OPTION TITLE='PRINT DATA FROM AN ASCII TAPE' DUMP VOL=ASCII,
// LRECL=80,RECFM=FB
TRANSLATE FROM=ASCII
More **TPUTIL** Examples

Tape TSTDT1 will be copied to tape BCKDT1. Then the labels (if any), the first two records, and the last 2 records of each file will be dumped to verify the backup operation.

**Figure 9. Using the **BACKUP** Command to Duplicate a Cartridge Tape.**

```
// jobname JOB (,,time,lines),'your name',CLASS=class
/*ROUTE PRINT node.location
/*SETUP TAPEC,2
// EXEC TPUTIL,
// DSN2= BACKUP VOL=(TSTDT1,BCKDT1) DUMP VOL=BCKDT1,STOP=2,LAST=2
```

This job will syntax-check all control statements (**OPT=EDIT**). No tapes will be mounted and **DSN1=NULLFILE** will prevent the allocation of any tape drives.

**Figure 10. Running a Syntax Check on **TPUTIL **Control Statements.**

```
// jobname JOB (,,time,lines),'your name',CLASS=class
/*ROUTE PRINT node.location
// EXEC TPUTIL,
// DSN1=NULLFILE OPT OPT=EDIT DUMP VOL=ANY,SKIP=(20,PICK=5),STOP=30,
// LAST=15,START=12,FILE=(3,7,10) BACK VOL=(TSTONE,TSTTWO),
// FILE=(1,5,7,9),OUTF=3 MAP VOL=TESTER,DDNAME=EXTRA
```

**Mapping an ASCII or Non-Labeled Tape**

The default for the **MAP** option is in EBCDIC format which is not good for a tape written in the ASCII format. How can you get around this restriction? Use the **DUMP** command combined with the **TRANSLATE** command to dump the first few records of each file on the tape. The **TRANSLATE** command is used to modify the execution of the **DUMP** command to make data recorded in other character codes easier to understand.

If your tape is a standard label ASCII tape, each file on the tape will be preceded by a header label and followed by a trailer label. These labels contain information about the data set including the record length and block size. If you dump the first few records of each file on the tape, with the **SELECT=ALL** parameter, even the labels are dumped as files. That is how you can pick up the **LRECL**, **BLKSIZE**, data set name and other information.

According to System 370/390 JCL by Gary D. Brown, the label file looks like this:

<table>
<thead>
<tr>
<th>Byte Position</th>
<th>No. Bytes</th>
<th>Contents (80-byte records)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>Characters 'HDR2' or 'EOV2' or 'EOF2'</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>Record format (F, V, etc.)</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td><strong>BLKSIZE</strong></td>
</tr>
<tr>
<td>11</td>
<td>5</td>
<td><strong>LRECL</strong></td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td><strong>DEN</strong></td>
</tr>
</tbody>
</table>

---

Utilities for Magnetic Tapes
This is not the complete table, but you may browse this book at the NERDC Support Desk [http://www.cns.ufl.edu/info-services/support/support_desk.html] (392-2061) if you do not have this reference.

If your tape is not a standard label tape (non-labeled), you would not be able to get the LRECL or BLKSIZE from this method or from a tape map. Both rely on getting the information from the label for each file. On a non-labeled tape, no information is stored on the tape to describe the format of the data.

For a non-labeled tape, use the keyword STOPAFTER= to specify 5 or more records to be dumped. Since the record length will be unknown, data will be read until the interblock gap is reached which indicates the end of a block. You will get 5 blocks dumped instead of records. However, you will also get a report of the BLKSIZE next to the printed block of data. If all of the blocks are the same size, you can make a good guess that the blocks are of fixed length.

Getting the record length is not as easy. Based on your own knowledge of the data, you can count/guess the number of bytes per record. Sometimes you can see repeating patterns in the printed block of data that can help you make a better guess. If you are still unable to determine the LRECL, you must contact the person who gave you the tape for this information.

This example dumps the first two records of each file from a reel tape (TAPE9). If the tape is a standard label ASCII tape, the first two records of each label will also be dumped. This is established by SELECT=ALL. The keyword STOPAFTER controls how many records are dumped.

**Figure 11. Mapping an ASCII or Non-labeled Tape**

```bash
//jobname JOB (,,time,lines),yourname,CLASS=class
/*SETUP TAPE9,1
/*ROUTE PRINT node.location
//**
//* PRINTS 2 RECORDS FROM EVERY FILE AND TRANSLATES FROM
//* ASCII INCLUDING LABELS IF PRESENT
//**
//MAP1 EXEC TPUTIL,UNIT1=TAPE9,VOL1=tapename
//SYSIN DD * DUMP VOL=tapename,SELECT=ALL,STOPAFTER=2 TRANSLATE FROM=ASCII
/*
```
TAPEDISK

Use the TAPEDISK procedure to copy a tape data set to a disk data set or to append a tape data set to the end of an existing disk data set. If the data set is created successfully, you can specify the disposition of the original tape data set.

The required parameters are the tape volume name (VOL) and, if the tape has standard labels, the tape data set name (TDSN). Using the default values for the other parameters will result in a primary allocation of 10 tracks and secondary allocations of 5 tracks for the disk data set. The record format, logical record length, and blocksize of the tape data set will be used for the disk data set. The old tape data set will be kept. The new disk data set will be cataloged if it is created successfully and will have the same name as the tape data set.

Note

If the tape data set name does not conform to EI&O conventions for naming disk data sets, you need to specify the disk data set name; otherwise, the new disk data set will be deleted at the end of the day. Do NOT specify TRECFM or TLRECL if copying a standard labeled tape.

JCL

Use the following JCL to access TAPEDISK:

Figure 1. JCL for Using the TAPEDISK Procedure

```
//jobname JOB (,,time,lines),'your name',CLASS=class
/*ROUTE PRINT node.location
/*SETUP TAPEC,1
// EXEC TAPEDISK,TDSN='name',VOL=volume,DDSN='name'
```

Parameters

You must specify the VOL parameter. If your tape is standard labeled, you must also specify TDSN; the other parameters are optional and have default values. Default values can be overridden by using any of the following parameters and values on the EXEC statement. Parameters must be separated from each other by a comma.

```
TDSN=name  (Required for an SL tape) specifies the tape data set name. The default is null.
TAPE=drive specifies a cartridge tape drive (, TAPE=TAPEC) at 38000 bpi, a 9-track tape drive at 6250 or 1600 bpi (, TAPE=TAPE9), or a 9-track tape drive with a density of 1600 or 800 bpi (, TAPE=TAPE8). The default is TAPEC.
```
FILE=file# specifies the relative position of the file on the tape. The default is 1.

LABEL=type specifies a non-standard tape label. The default is SL.

TRECFM=format specifies a tape record format. The default is null.

TLRECL=length specifies a tape logical record length. The default is null.

TBLKS=size specifies a tape blocksize. The default is null.

PRIM=tracks specifies a primary track allocation. The default is 10.

SEC=tracks specifies a secondary track allocation. The default is 5.

DRECFM=format specifies the disk record format. The default is the value of TRECFM.

DBLKS=size specifies a disk blocksize. The default is the value of TBLKS.

DDSN=name specifies a disk data set name. The default is the value of TDSN.

DISP=disp specifies the disposition (PASS or DLTE) of the tape data set upon successful job completion. The default is KEEP.

RETAIN=RETAIN specifies that the tape not be dismounted at the end of the job step. The default is null.

N1DISP=disp specifies disposition of the disk data set at the start of the job step. The default is NEW.

N2DISP=disp specifies disposition of the disk data set at the end of the job step. The default is CATLG.

UNIT=unit specifies type of disk unit. The default is SYSDA.

DVOL=volume specifies the name of a specific disk volume. The default is null.

Examples

The following example shows how to create a disk data set called UF.STARTRK.PHASER from the tape file whose name is FILE1. The volume name is TAPE01. Since the tape file name does not conform to EI&O data set naming conventions, using the DDSN parameter allows you to specify a name for the disk data set.

// EXEC TAPEDISK,TDSN='FILE1',VOL=TAPE01,
// DDSN='UF.STARTRK.PHASER'

The sample job shown in Figure 16 creates a disk data set called UF.NTERPRIZ.TEN.FORWARD from file 3 of a non-labeled, 9-track tape called MYTAPE.
The tape is non-labeled and contains 80-byte records. Since the tape does not have labels, you must specify the logical record length (\texttt{TLRECL}) and block size (\texttt{TBLKS}) of the tape data set, as well as a disk data set name. Note also that the disk block size is changed.

\textbf{Figure 2. Example to Copy a File from a 9-Track Tape to Disk.}

\begin{verbatim}
/*ROUTE PRINT node.location
/*SETUP TAPE9,1
// EXEC TAPEDISK,VOL=MYTAPE,TAPE=TAPE9,LABEL=NL,
// FILE=3,TLRECL=80,TBLKS=800,
// DDSN='UF.NTERPRIZ.TEN.FORWARD',DBLKS=6160

To append a tape data set named BOOKS to the end of a disk data set named UF.RESEARCH.BOOKS, code the \texttt{EXEC} statement as follows:

// EXEC TAPEDISK,VOL=MYTAPE,TDSN=BOOKS,
// DDSN='UF.RESEARCH.BOOKS',
// N1DISP=MOD
\end{verbatim}