B I MOYLE ASSOCIATES, INC.
5788 LINCOLN DRIVE • MINNEAPOLIS, MN 55436 • 612-933-2885

BIM-Dr.D

(also known as 'Doctor D')

DISK UTILITY SYSTEM

USERS GUIDE

Release 6.5.4

This documentation applies to Release 6.5.4 of the program product BIM-Dr.D.

Original Printing.....09/04/98
Last Revised 09/27/99

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

Introduction

Overview

BIM-Dr.D, also known as Doctor D, is referred to throughout this manual as Dr.D.

Dr.D is a fast and flexible disk utility system. It provides the fastest backup and restore functions currently available.

A very flexible package, Dr.D can be used to satisfy the needs of users in the following situations:

- Disaster recovery backup with standalone / VM restore
- Disk compacting
- Library backup and reorganization
- Convert files and libraries to different disk types
- VSAM / ISAM / PSAM reorganize
- Test file creation
- VSAM listcat, file definition, file delete and verify
- Tape copy and tape test
- Disk manager pool reorganization / defragmentation
- Disk manager pool backup / restore
- Individual and pool list VTOC

Dr.D is especially useful and fast for a wide range of SAVE, RESTORE, and COPY functions. Dr.D can:

BACKUP - Dr.D allows you to save any volume, logical disk file, or library to tape or disk file.

RESTORE tape to disk- Dr.D allows you to restore any volume, logical disk file, or library from tape (or disk file) to disk.

COPY from disk to disk- Dr.D allows you to copy any volume, logical disk file, or library from disk to disk.

Dr.D manipulates the three entities most needed of a disk utility: disk volumes, data files, and libraries. In addition, it also provides a number of miscellaneous functions.

Volumes: Dr.D SAVES, RESTORES, or COPIES the entire or just active portions of a disk. It can also create IPL records on the save tape so the tape may later be IPLed and restored in a standalone fashion or under VM.

Data Files: Dr.D Saves, Restores, or Copies entire logical disk files. These files may be VSAM, direct access, sequential or indexed sequential, multivolume, multiextent, contain user labels, be split cylinder, or any combination thereof. For maximum flexibility, the restored or copied file does not need to be restored to the same disk location; it need not be the same size; it need not contain the same number of extents; and it need not reside on the same type of disk as the saved file. Moreover, a DA file may be restored or copied as an SD and vice versa. VSAM or ISAM files will be automatically reorganized upon being restored or copied.

Libraries: Dr.D SAVES, RESTORES, or COPIES all DOS libraries. Libraries may be System or Private-Source, Relocatable, Procedure, or Core. The restored or copied library does not need to be restored to the same disk location; it need not be the same size; nor need it reside on the same disk type as the saved library. A system library may be restored or copied as a private library and vice versa. Also, a library may be restored in a manner so as to add its contents to an already existing library, again without regard to it being a system or private library. All libraries are automatically condensed upon being restored or copied. Dr.D offers a reorganize feature which alters SYSRES allocations just prior to restoring system libraries.

Miscellaneous: Dr.D offers several features for handling a variety of tasks.

These include:

- Clear Disk- all or a portion
- Create a Format-1 Label
- SAVE, RESTORE or COPY a physical extent. Specify starting and ending cylinder and head to delimit the extent. As always, the restored or copied extent doesn't have to 1) be restored or copied to the same location, 2) be the same size, or 3) even reside on the same type of disk as the saved extent.

Examples

A wide range of examples are provided to cover most SAVE/RESTORE/ COPY situations. The user should note, however, that these examples and JCL statements are for illustrative purposes only. Users are expected to modify them as required by their specific requirements and environment.

Compatibility

Dr.D is compatible with any IBM system ES90xx, 30xx, 43xx, or plug compatible CPU operating under the VSE-ESA, VSE-SP, VSE, or DOS-MVT-VSE operating systems, and any disk or tape device supported by those operating systems.

Job Control Language Examples

The examples in the following section illustrate the JCL setup for Dr.D. Note: For specific JCL follow the examples.

```
// JOB NONAME
// ASSGN SYS007,...         (01)
// TLBL TAPETWO
// TLBL TAPE003
// TLBL TAPEOUT             (02)
OR
// DLBL TAPEOUT
// EXTENT SYS......
// ASSGN SYS008,...         (03)
// ASSGN SYS006,...
// ASSGN SYS005,...
// TLBL TAPEIN              (04)
OR
// DLBL TAPEIN
// EXTENT SYS......
// DLBL ANYFILE,'ETC.'       (05)
// EXTENT ETC.               (06)
// EXTENT ETC.               (07)
// EXEC DRD,SIZE=200K        (08)
CONTROL STATEMENTS           (09)
/*                           (10)
/&                           (11)
```

General Notes:

If neither SAVING nor RESTORING, steps 01-04 may be omitted.

Dr.D reads and writes to the SYSLOG console. It reads control cards from any SYSIPT unit record card reader, tape drive or disk extent. SYSIPT data may also be included in a procedure library.

Dr.D control statements may be printed on log or list (see 'LST' keyword in keyword section and 'UPSI' settings).

Specific Notes:

(01,02) These statements are not required if you are not saving.

If a save is required in this job step, either a DLBL or TLBL for TAPEOUT must be provided.

If a TLBL is used:

SYS007 must be assigned to the save tape. There is one standard label tape file TAPEOUT created per execution of Dr.D, regardless of the number of files, packs, libraries, etc. saved during this run. If multiple save tapes are to be created simultaneously, see the section below.

If a DLBL is used:

A DLBL and EXTENT must be provided to define a disk file large enough to contain the data to be saved. If an FBA disk is used, no CISIZE or BLOCK SIZE parameter is needed on the DLBL. If a CKD disk is used, the BLOCK SIZE parameter should be used to set the disk blocks to an efficient size (use track length if possible).

(03,04) Unless you are restoring, these statements are not required.

However, if a restore is required in this job step, then either a TLBL or DLBL for TAPEIN is necessary.

If a TLBL is used:

SYS008 must be assigned to the restore tape. This standard label tape's file name is TAPEIN.

If a DLBL is used:

A DLBL and EXTENT must be provided matching that used during backup with the appropriate BLOCK SIZE DLBL parameter. (05-07) Use DLBLS and EXTENTS as required. Note: That DLBLs and EXTENTS may reside as a system or partition standard label and in that case are not needed in the JCL itself. The 'SYSNO' of the EXTENT card is not required if the 'VOLSER' is present.

(08) Execute Statement. For VSE type systems, ',SIZE=' should be used to insure both adequate partition size for tables, buffers, etc., while leaving enough memory for functions requiring 'GETVIS' space (VSAM, FBA library copy/backup). Generally, 100K will allow most functions to be performed, using the 16K tape block size, or 200K if the maximum 65K tape block size is used.

NEVER EXECUTE Dr.D WITH SIZE=AUTO

(09) This is the place for Dr.D control cards. These cards specify what actions are to be taken by the programs. Any number of permutations of saves, copies, or restores may be defined. (For more information on Control Cards see "Appendix VI - DOS/VSE OPERATIONS" on page 205).

(10) End of data to Dr.D control cards. Note that if file or member selection is used (MEMBER/AREA/PARTIAL=SELECT/REJECT) that an additional /* is required to terminate the selection process (/@ is also acceptable for selection termination).

Creation of Multiple Copies of Backup Tapes

Up to three tapes can be created simultaneously- production of the second tape is controlled by setting the 'UPSI' switch 3 (UPSI xxx1) in either the JCL UPSI or by the Dr.D UPSI= keyword. Since all eight UPSI switches have been given Dr.D functions, it was necessary to add a ninth UPSI to designate the production of the third tape. This ninth UPSI is a pseudo UPSI and must be entered on the Dr.D control card (UPSI=XXX1XXXX1), not by the JCL UPSI statement. The third tape TLBL/DD name is TAPE003.

Example - Create multiple copies of a single backup tape.

```
// JOB BACKUP TO THREE TAPES
// TLBL TAPEOUT,'FIRST TAPE OUT'
// TLBL TAPETWO,'SECOND TAPE OUT'
// TLBL TAPE003,'THIRD TAPEOUT'
// ASSGN SYS007,180  TAPEOUT
// ASSGN SYS006,181  TAPETWO
// ASSGN SYS005,182  TAPE003
// EXEC DRD,SIZE=200K
SAVE=NO,UP=111100001,PRINT=LIST,CP=YES,BLOCK=65496
SAVE=CURRENT,SV=DOSRES,EOF=YES
SAVE=NO,TAPE=RUN
/*
```

The 'TD=' keyword (Tape Ddname) keyword can be used to allow the TAPEOUT/TAPETWO/TAPE003 names to be varied. All three names can be specified- TD=TAPEOUT/TAPETWO/TAPE003, or any combination as long as the '/' is present- TD=//TAPEXXX for instance would specify the name for the third tape while TD=/TAPEYYY would specify the name for the second tape. If the TD= keyword is used to specify the name for either TAPETWO or TAPE003, then the UPSI setting is not needed to invoke the creation of the additional tape(s). Invoking the creation of the third tape always implies creation of the second tape, i.e. specifying UPSI=XXXXXXXX1 is equivalent to UPSI=XXX1XXXX1 and specifying TD=//TAPE003 is the same as TD=/TAPETWO/TAPE003.

How to Save, Restore, or Copy

To SAVE disk files to tape, specify 'SAVE=' on the control card.

To RESTORE files back to disk, specify the 'RESTORE=' parameter.

In some cases, SAVE or RESTORE is used when no user data is involved. For example, 'SAVE=IPL' is used to place the Dr.D standalone supervisor on a backup tape, 'RESTORE=ERASE' is used to specify that areas of disk are to be cleared, and 'RESTORE=PUTLABEL' is used to put a label in a VTOC.

The 'SAVE/RESTORE=NOPERATION' command is used to tell Dr.D that there will be no save/restore operation performed. The 'SA/RE=NOP' statement is used before and after the actual save/restore operation, on it Global Keywords are specified such as block size, upsi bits, tape compression and print options just to name a few. On the second 'NOP' statement the tape operation to be performed is specified as 'TA=REwind, TA=NORewind, or TA=RUn (unload)'. This is usually followed by a (/*) and (/&) which terminates the Dr.D execution. The 'NOPERATION' command like other commands in the manual may be abbreviated to 'NOP' or 'NO' to prevent lengthy JCL coding.

In order to accomplish a Copy (Disk to Disk) simply list all the operands necessary to perform a SAVE and a RESTORE on the same control card.

Example 1. Save Disk to Tape

```
SAVE=CURRENT,SVOLSER=111111,EXPIRED=YES
```

Example 2. Restore to Disk

```
RESTORE=CURRENT,RVOLSER=111111
```

Example 3. Copy Disk to Disk (No Tape Involved)

```
SAVE=CURRENT,RESTORE=CURRENT,SVOLSER=111111,RVOLSER=222222,
EXPIRED=YES
```

Note: Whenever a SAVE and RESTORE operand are identical, the COPY operand may be substituted to save time in filling out control cards.

Example 4. Using COPY to reduce parameter coding.

SVOLSER=111111,RVOLSER=111111 is equivalent to CVOLSER=111111
SBEGIN=43210,RBEGIN=43210 is equivalent to CBEGIN=43210
SAVE=CURRENT,RESTORE=CURRENT is equivalent to COPY=CURRENT

The control card:

SAVE=CURRENT,RESTORE=CUR,SVOLSER=111111,RVOLSER=111111,EXPIRED=YES

could be shortened to:

COPY=CURRENT,CVOLSER=111111,EXPIRED=YES

Dr.D CICS Interface (Dr.D/ON-LINE)

Dr.D provides an interactive interface (Dr.D/ON-LINE) through CICS/VSE.

Dr.D/ON-LINE allows the user to generate job streams that will execute Dr.D in a batch environment. The user can control the type of job stream to be built and tailor the job control for their specific environment. Most Dr.D batch facilities are available through the Dr.D/ON-LINE system. Dr.D/ON-LINE also provides centralized processing for other interactive online Dr.D facilities. A main menu panel allows the user to not only select Dr.D/ON-LINE processing, but also the online DRZAP, the new improved DRDITTO, and various other online services.

How Dr.D/ON-LINE Works

Dr.D/ON-LINE is a set of programs written in CICS command level code that interactively builds job streams to perform batch Dr.D services. The Dr.D/ON-LINE programs use the CICS basic mapping facility to perform terminal displays and interact with the user.

Dr.D/ON-LINE provides panels that the user can fill out to provide information for the Dr. D job stream to be built. The user can specify the I/O device information, the VSE and POWER job information, as well as the Dr.D function to be provided through these panels.

The Dr.D/ON-LINE interface does not actually perform the Dr.D function, it builds a job stream that can be submitted to a batch partition for execution. The job stream created can also be pulled into an online editor and used as a skeleton for future Dr.D processing.

Dr.D/ON-LINE Operating Procedures

Once you have updated CICS for the Dr.D/ON-LINE transaction, programs, and maps, you can activate the Dr.D/ON-LINE interface.

Running Dr.D/ON-LINE as a Transaction

A single transaction called 'DRD' will activate the CICS interface for Dr.D/ON-LINE. Enter the 'DRD' transaction ID to bring up the main menu panel and choose the Dr.D service to be performed. Program function key 3 will get you out of the current panel and terminate the transaction from the main menu panel.

Diagnostics and errors that occur during the Dr.D/ON-LINE interactive processing will be displayed on the terminal, usually at line 24. The Dr.D/ON-LINE messages are documented in the Dr.D user manual and are assigned the range of 401 through 449. Transaction abends and dumps may accompany the diagnostic messages to provide useful information for system debugging. The Dr.D/ON-LINE transaction dumps can be identified via 'DRDn'. These dumps only occur when a message will not suffice and should be sent to your Dr.D technical support department for further action.

Running Dr.D/ON-LINE under VSE Interactive Interface

Once you have setup Dr.D/ON-LINE to run under the VSE Interactive Interface you may activate the Dr.D/ON-LINE interface via a VSE Interactive Interface panel. When the Dr.D/ON-LINE interface has completed, control will be returned to the VSE interactive interface panel.

Installation

Dr.D is shipped on an unlabeled tape or cartridge containing a POWER offload of a RDR queue job. When this tape is loaded into your RDR Queue, there will be one POWER job (INSTALL) which will catalog Dr.D to your libraries.

Step 1 - Load Distribution Tape

Mount the distribution tape on a tape drive and enter the following POWER/VSE command on the console:

```
S RDR,cuu
```

where “cuu” is the address of the tape drive.

Step 2 - Release Install Job

Alter the POWER job (INSTALL) that was loaded in the RDR Queue to run in a partition that has labels established for link editing.

```
A RDR,INSTALL,DISP=D,CLASS=?
```

Step 3a - Installation in VSE/SP and VSE/ESA

When job INSTALL starts up, it will pause allowing you to enter required JCL to catalog Dr.D to your VSE/SP or VSE/ESA system.

Note that the required responses are in lower case (BG partition '0' assumed).

```
// JOB INSTALL  DR.D
*         THIS IS  DR.D VER-X.X.X EXPIRATION DATE-YY/DDD
// PAUSE ENTER 'TO' LIBDEFS FOR CL
0 libdef *,catalog=vsesp.drdlib
0 (HIT ENTER AFTER THE LIBDEF IS ACCEPTED- THEN WAIT FOR THE PAUSE)
*  THEN ENTER: EXEC LIBR,PARM='ACCESS SUBLIB=LLLLLLL.SSSSSSSS'
// PAUSE   ENTER COMMANDS
0 exec libr,parm='access sublib=vsesp.drdlib'
0 input sysipt
0 (HIT ENTER)
// PAUSE CATALOGING COMPLETE, APPLY ANY FIXES NOW
0
// PAUSE DR.D INSTALL COMPLETED
0
// PAUSE TYPE CANCEL TO BYPASS MANUAL PRINTING
cancel bg
EOJ INSTALL
```

Step 3B - Installation in VSE and MVT/ESA (not VSE/SP or VSE/ESA)

When job INSTALL starts up, it will pause allowing you to enter required JCL to catalog Dr.D to your VSE or MVT system.

Note that the required responses are in lower case (BG partition '0' assumed).

```
// JOB INSTALL  DR.D
*         THIS IS  DR.D VER-X.X.X EXPIRATION DATE-YY/DDD
// PAUSE ENTER 'TO' LIBDEFS FOR CL
0 libdef cl,to=usrcl1
0 (HIT ENTER AFTER THE LIBDEF IS ACCEPTED- THEN WAIT FOR THE PAUSE)
*  THEN ENTER LIBDEF FOR RELO, THEN ENTER: EXEC MAINT
// PAUSE   ENTER COMMANDS
0 libdef rl,to=usrrl1
0 exec maint
0 (HIT ENTER)
// PAUSE CATALOGING COMPLETE, APPLY ANY FIXES NOW
0
EOJ INSTALL
```

Step 4 - Dr.D Example Library Installation

A number of example job streams are contained on the Dr.D installation tape. These should be loaded into an editor (ICCF, CMS, BIM-EDIT, etc.) for reference and use. This can be accomplished by first positioning the Dr.D installation tape to the 5th file (// MTC FSF,180,4), and then punching the tape to the punch queue using the DITTO function 'TC'. The entry in the punch queue can then be extracted by the editor of your choice. This tape file contains 80 byte records blocked 200.

Step 5 - Preparing CICS For Dr.D/ON-LINE

You must update your CICS system before you can utilize Dr.D/ON-LINE. Entries must be made to the CICS PPT and PCT for the Dr.D/ON-LINE programs, BMS maps, and the transaction ID (DRD).

Dr.D/ON-LINE utilizes the basic feature of the CICS Basic Mapping Support (BMS). Dr.D/ON-LINE also uses the CICS SPOOL interface which is generated via the SIT SPOOL specifications. No additional CICS support is required for Dr.D/ON-LINE.

Step 5A - Updating The CICS PCT

Add the following copy statement to your PCT source, reassemble and catalog it:

```
COPY DRD$PCT
```

Step 5B - Updating The CICS PPT

Add the following copy statement to your PPT source, reassemble and then catalog it:

```
COPY DRD$PPT
```

Step 5C - Adding Dr.D/ON-LINE to the VSE Interactive Interface

The Dr.D/ON-LINE interface will also support the VSE Interactive Interface. The following steps illustrate the changes required to implement the Interactive Interface support for Dr.D/ON-LINE:

1. Activate the Interactive Interface
2. Select the 'Resource Definition' panel
3. Select the 'User Interface Tailoring' panel
4. Add Dr.D/ON-LINE selection via 'Maintain Selection Panels':

   Name=DRDCICS
   Type=1
   Text='Dr.D/ON-LINE Interface'

5. Add Dr.D/ON-LINE selection via 'Maintain Application Profiles':

   Name=DRDCICS
   Activate=DRD
   Code=1
   Case=1
   Data=IESDRD$
   Show=2

VSAM Data File Functions

VSAM Catalog Logical Feature

The Dr.D VSAM LOGICAL feature is designed to enable some or all of the objects within a 'VSAM' catalog to be saved or restored with a single Dr.D statement. At the same time, the feature allows individual objects to be selected for restoration, if desired.

Tapes produced by this feature are fully compatible with all Dr.D functions. You can combine other Dr.D saves such as volumes or libraries on the same tape and you can use the 'FIND' and 'SKIP' parameters in the usual way. To help you, Dr.D will print a list of the files that it saves, if the print options are set.

Example 1. Save contents of two catalogs.

This example saves the contents of two VSAM catalogs to tape.

```
// JOB VSAM LOGICAL SAVE
// ASSGN SYS007,TAPE
// TLBL TAPEOUT
// DLBL UCAT1,'VSAM.USER.CATALOG.ONE',,VSAM
// DLBL UCAT2,'VSAM.USER.CATALOG.TWO',,VSAM
// EXEC DRD,SIZE=200K
SAVE=NOP,BL=65496,UPSI=111,CP=YES SET TAPE BLOCKSIZE & UPSI
SAVE=LOGICAL,SCAT=UCAT1,FAST=YES   SAVE FILES IN UCAT1
SAVE=LOGICAL,SCAT=UCAT2,FAST=YES   SAVE FILES IN UCAT2
SAVE=NO,TAPE=RUN           UNLOAD THE TAPE AT COMPLETION
/*
/&
```

The example above will save all objects defined in each catalog in alphabetical order of name within catalog. 'DLBL' statements are not required for the VSAM files being saved.

It is possible to be much more selective about which objects are saved by adding more parameters to the 'SAVE=LOGICAL' statement.

'TSTAMP=XXXXX' can be used to select files whose time stamp is greater than, less than, or equal to the date-time specified. See section "Appendix I - Summary of Dr.D Keywords" on page 172 for a full explanation of this parameter.

The 'PARTIAL=' parameter allows selection or rejection of objects based on all or part of the object names. It also allows file selection based on the file condition (see section "VSAM TUNING" on page 35).

```
PARTIAL=SELECT
PARTIAL=REJECT
```

If either of these keywords is used, the 'SAVE=LOGICAL' statement is followed by one or more statements containing names of objects to be selected from the 'VSAM' catalog.

One name is specified per statement, contained in positions one to 44, or if a string search is desired, '=S=' is entered in positions one to three, followed by the search argument. If the phrase '@REVERSE@' is added after the select parameters, with one or more spaces separating, the selection result is reversed. That is, if 'PA=SE' is in effect, a match will be rejected. If 'PA=RE' is in effect, a match will be selected. In both cases, the order of entry of selector cards determines which takes effect.

If a selection name is entered that begins with "%" or "%%", then the name is expanded in accordance with the VSAM definition of partition/cpu dependent files.

All objects in the catalog are compared against all specified names. Only those objects which are selected are saved.

Additional selection is allowed by use of the 'SVOLSER=' parameter to limit the selection process to those files that fully or partially reside on a particular disk. Two facilities are available to allow 'GENERIC' specification of names.

A '/' character in any position is a 'wild' character. Dr.D does not examine that position in selecting a name.

A '*' character indicates that the remainder of the specification is 'wild' - any name matching up to the character to the left of the '*' will be selected without further checking.

The following examples illustrate this point:

/ / / / / / / /VSAM.WORK.FILE

Selects any object whose name ends with the characters 'VSAM.WORK.FILE' starting in position 9.

PAYROLL*

Selects any object whose name starts with the characters 'PAYROLL'.

/ / / / / / / /PARTS.LIST*

Selects any object whose name has the characters 'PARTS.LIST' in positions 9 to 18.

=S=PAYROLL

Selects any object where 'PAYROLL' occurs in any ID position.

The 'PARTIAL=REJECT' keyword can be used to produce the opposite effect. All objects are saved unless their names match the selection criteria.

The 'VSAM' automatic definition feature is always invoked for all objects saved by a 'SAVE=LOGICAL' statement, to save 'VSAM' catalog definitions along with the data.

Users may specify what Dr.D will do when it finds 'VSAM' object in use in another partition, by means of the 'VUSE=' keyword.

VUSE=NULL	WILL BACKUP A FILE AS NULL
VUSE=IGNORE	BYPASS ANY FILE THAT IS OPEN
VUSE=CANCEL	CANCEL THE JOB
VUSE=OPERATOR	LET OPERATOR SPECIFY OPTION
VUSE=SAVE	SAVE THE FILE ANYWAY (FAST=YES REQUIRED)

Example 2. Selective VSAM logical save.

This example saves all payroll and accounts files from 'VSAM.USER.CATALOG' but does not save any files which have expired. It ignores 'ACCOUNTS.RECEIVE.ABLE*'. It also selects 'GL.TRAN.FILE' based on its DLBL.

```
// JOB VSAM LOGICAL SAVE - SELECTIVE
// ASSGN SYS007,TAPE
// TLBL TAPEOUT
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// DLBL SELECTD,'GL.TRAN.FILE',,VSAM
// EXEC DRD,SIZE=200K
   SAVE=NOPERATION,UPSI=111,BLOCKSIZE=65496
   SA=LOGICAL,SCAT=IJSYSUC,PARTIAL=SELECT,EXPIRED=NO,FAST=YES
   PAYROLL*
   ACCOUNTS.RECEIVE.ABLE*        @REVERSE@
   ACCOUNTS*
   DD=SELECTD
   /@
   SAVE=NOP,TAPE=RUN
/*
/&
```

VUSE=CANCEL (Default)

This is the default action of 'VUSE=' parameter. Dr.D cancels the whole job after issuing the message below.

DOCTOR 175 DATASET 'FILE-ID' IN USE - SAVE RUN CANCELED

If 'VUSE=' is specified, the following options are available.

VUSE=IGNORE

If 'VUSE=IGNORE' is specified, Dr.D will completely ignore any object which is in use. The message below then appears on the console:

DOCTOR 175 DATASET 'FILE-ID' IN USE - SAVED AS NULL

VUSE=NULL

Dr.D will save the object as a null file without issuing any message.

VUSE=SAVE

If 'FAST=YES' is specified, the file is saved, but results may be unpredictable if the file is being updated by another task.

VUSE=OPERATOR

Dr.D requests that the operator enter the VUSE option. If the 'VSAM' file is null (no data) it will be saved as such unless the 'VNULL=' parameter indicates a different action.

VNULL=YES/NO

Unless 'VNULL=NO' is specified, Dr.D will save null files. If null files are not to be saved Dr.D issues the following message.

DOCTOR 176 - DATASET - 'FILE-ID' EMPTY - IGNORED

The VSAM catalog feature (unless otherwise instructed) saves all objects within a catalog in alphabetical order, but does not save any 'PATH'. Dr.D may be instructed to recognize the associations between 'VSAM' objects such as 'PATHS and ALTERNATE INDEXES' by specifying the following parameter.

ASSOC=YES

This causes Dr.D to save 'ALTERNATE INDEX' clusters following their associated base clusters, and 'PATHS' following the clusters to which they relate. Thus, on restoring, the various types of object will be restored in the correct order.

When the 'PARTIAL=' keyword is also specified, 'ASSOC=YES' causes Dr.D to apply selection or rejection criteria to the base clusters only. If a base cluster is selected or rejected, all associated clusters will also be selected or rejected, regardless of whether or not the names of these associated clusters match the selection criteria.

When restoring associated clusters, 'ESDS' base clusters must be restored with the same 'CISIZE' and 'RECSZ' definitions as when they were saved to ensure that any alternate index remains valid.

Example 3. VSAM logical save with associations.

In this example, assume that all application files are defined in the 'VSAM' master catalog. This example saves all the files together with any associated 'ALTERNATE INDEXES' and 'PATHS'.

```
// JOB VSAM LOGICAL SAVE WITH ASSOCIATIONS
// ASSGN SYS007,TAPE
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
   SAVE=NOP,BLOCKSIZE=65496,UPSI=111,PRINT=YES
   SAVE=LOG,SCAT=IJSYSCT,ASSOC=YES,VUSE=CA,VNULL=IGNORE,FAST=YES
   SAVE=NOP,TAPE=RUN
/*
/&
```

Example 4. VSAM logical restore.

The 'RESTORE=' operation is almost a direct reversal of the 'SAVE=' operation, as this example illustrates. It restores the contents of the two 'VSAM' catalogs saved in Example 3. The 'VSAM' catalog into which you restore is determined by the 'RCAT=' parameter and its label.

```
// JOB VSAM LOGICAL RESTORE
// ASSGN SYS008,TAPE
// TLBL TAPEIN
// DLBL UCAT1,'VSAM.USER.CATALOG.ONE',,VSAM
// DLBL UCAT2,'VSAM.USER.CATALOG.TWO',,VSAM
// EXEC DRD,SIZE=200K
   RESTORE=NOPERATION,BLOCKSIZE=65496,UPSI=111
   RESTORE=LOGICAL,RCATDDNAME=UCAT1
   RESTORE=LOGICAL,RCATDDNAME=UCAT2
   RESTORE=NOP,TAPE=RUN
/*
/&
```

The 'PARTIAL=' parameter has the same meaning for 'RESTORE=' as for 'SAVE='.

The 'VUSE=' parameter has no meaning during restore and will be ignored if specified. If an attempt is made to restore to a 'VSAM' file in use, the restore will be canceled.

The 'VNULL=' parameter also has no meaning on restore and will be ignored. If a null object was saved it will be restored as a null object.

The 'ASSOC=' parameter has a slightly different meaning when restoring data.

`ASSOCIATIONS=NO`

Any 'PATHS' found on the tape will be ignored and not restored.

`ASSOCIATIONS=YES`

All 'PATHS' will be restored when the 'ASSOC=YES' is specified. If you save the contents of more than one catalog to the same tape, you need some method of identifying which catalog you wish to restore.

When you do a 'SAVE=LOGICAL', Dr.D saves a catalog header record before any clusters from the catalog. This record is identified by the 44 character file identifier provided in the label information for the catalog identified by the 'SCAT=' parameter.

On a 'RESTORE=LOGICAL', Dr.D uses the 'RCAT=' parameter to access the label information for the catalog into which you are restoring, and scans the tape forward from the present position to locate the matching catalog header record. If the saved catalog had a different name from the restored catalog, then the saved catalog is identified on the 'RESTORE=LOGICAL' statement by using the 'SCAT=' parameter.

Example 5. VSAM catalog selective restore.

This example is similar to Example 4, except it bypasses the restore of 'VSAM.USER.CATALOG.ONE', and restores the saved contents of 'VSAM.USER.CATALOG.TWO' into another catalog called 'NEW.VSAM.USER.CATALOG'.

```
// JOB VSAM CATALOG SELECTIVE RESTORE
// ASSGN SYS008,TAPE
// TLBL TAPEIN
// DLBL NEWCAT,'NEW.VSAM.USER.CATALOG',,VSAM
// DLBL UCAT2,'VSAM.USER.CATALOG.TWO',,VSAM
// EXEC DRD,SIZE=200K
   RESTORE=NOPERATION,BLOCKSIZE=65496,UPSI=111
   RESTORE=LOGICAL,SCATDDNAME=UCAT2,RCATDDNAME=NEWCAT
   RESTORE=NOP,TAPE=RUN
/*
/&
```

File Rename Feature

During restore, file IDs may be altered by several methods, in conjunction with the 'PARTIAL=SE/RE' parameter. The IDs may be fully replaced, transformed, prefixed, or any combination of these functions, by the addition of rename control cards following the selector card used to select the file for restore.

The three possibilities are as follows:

1. Starting in column 1 code '=T=' followed by a search string an '=' and a replace string (or space if the search string is to be removed). Only the first occurrence of the string is replaced, so use multiple '=T=' cards if you wish to transform multiple occurrences.
2. Starting in column 1 code '=P=' followed by a string to be added to the front of the generated ID.
3. Code the replacement ID starting in column one and '@RENAME@' to the right of the replacement (separated by at least one space). Use the '*' to halt character replacement, or the '/' to prevent replacement of individual characters.

Note: That the selection and rename process is applied to the file associations as well as to each file restored. If file 'CUR.STAT.FILE' is transformed to 'TEST.CUR.STAT.FILE' and it has an alternate index named 'CUR.STAT.FILE.AIX', then the associated file is named 'TEST.CUR.STAT.FILE.AIX'. If the associated file is not selected, then the file define may fail.

Example 6. VSAM selective restore with Rename.

This example restores certain files from the backup tape, and for test purposes, renames them. At the same time, it standardizes the names of 'EMPLOYEE.PRMASTER', 'YTD.PRMASTER', and 'QTD.PRMASTER', so that each file starts with 'PAYROLL.MASTER'.

```
// JOB VSAM LOGICAL RESTORE WITH RENAME
// ASSGN SYS008,TAPE
// TLBL TAPEIN
// DLBL UCAT1,'VSAM.USER.CATALOG.ONE',,VSAM
// DLBL UCAT2,'VSAM.USER.CATALOG.TWO',,VSAM
// EXEC DRD,SIZE=200K
   RESTORE=NOP,BLOCKSIZE=65496,UPSI=111
   RESTORE=LOGICAL,RCATDDNAME=UCAT1,PARTIAL=SELECT,ASSOC=YES
   CUR.STAT.FILE*
   =P=TEST.
   =S=PRMASTER
   =T=.PRMASTER=
   =P=PAYROLL.MASTER.
   /@
/*
/&
```

Note: That the word 'PRMASTER' may appear anywhere in the ID and that the '=T=' card removes it, followed by prefixing of 'PAYROLL.MASTER.'.

CAUTION: If the ID generated by the rename process matches an existing file in the target catalog, that file will be replaced by the renamed file. If you are unsure of the outcome, restore to a test catalog first.

Example 7. VSAM save selecting with 'SVOLSER='

Use of the 'SVOLSER=' parameter allows files within a catalog to be selected if they fully or partially reside on a particular disk.

```
// JOB VSAM SELECTION BY VOLUME SERIAL NUMBER
// ASSGN SYS007,TAPE
// TLBL TAPEOUT
// DLBL UCAT2,'VSAM.USER.CATALOG.TWO',,VSAM
// EXEC DRD,SIZE=200K
   SAVE=NOPERATION,BLOCKSIZE=65496,UPSI=111
   SAVE=LOGICAL,SCAT=UCAT2,SVOL=VSAM01,ASSOC=YES,PARTIAL=SELECT
   *
   /@
   SAVE=NOP,TAPE=RUN
/*
/&
```

Example 8. Restore one disk of a full catalog.

A full catalog backup may be restored so that only files that fully or partially resided on a particular disk (at save time) are restored.

```
// JOB VSAM SELECTION BY VOLUME SERIAL NUMBER
// ASSGN SYS008,TAPE
// TLBL TAPEIN
// DLBL UCAT2,'VSAM.USER.CATALOG.TWO',,VSAM
// EXEC DRD,SIZE=200K
   RESTORE=NOP,BLOCKSIZE=65496,UPSI=111
   REST=LOGICAL,RCAT=UCAT2,SVOL=VSAM01,ASSOC=YES,PARTIAL=SELECT
   *
   /@
   RESTORE=NOP,TAPE=RUN
/*
/&
```

Unique File Backup/Restore

VSAM 'UNIQUE' files can be saved/restored by either the catalog driven 'SAVE/RESTORE=LOGICAL' function or the 'SAVE/RESTORE=VSAM' function. In either case, additional DLBL and EXTENT JCL statements are required in order for the file to be defined. If the 'RESTORE= LOGICAL' function is used the generated DTF/DDNAME is altered replacing the first character. Thus the name is changed from 'FILnnnn' to 'DILnnnn' and 'XILnnnn'. If the 'RESTORE=VSAM' function is used, the first character of the 'RFILE=' name is changed in the same way (do not start the 'RFILE=' name with 'D' or 'X').

Example 9. Unique File backup using 'SAVE=LOGICAL'.

This example shows the backup of a UNIQUE file from a VSAM Catalog.

```
// JOB V9 UNIQUE FILE BACKUP
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
   SAVE=NOP,UPSI=111,BLOCKSIZE=65496,PRINT=YES,WS=MCATPASS
   SAVE=LOGICAL,SCAT=IJSYSCT,PARTIAL=SELECT,ASSOC=YES,FAST=YES
   UNIQUE.KSDS.FILE
   /*
   SAVE=NOP,TAPE=REW
/*
/&
```

Example 10. Unique File restore using 'RESTORE=LOGICAL'.

This example shows the Restore of a Unique file.

```
// JOB V10 UNIQUE FILE RESTORE USING RESTORE=LOGICAL
// TLBL TAPEIN,'TAPEOUT'
// DLBL DIL0001,'UNIQUE.KSDS.FILE',,VSAM,CAT=IJSYSCT
// EXTENT ,FBA341,1,0,26982,1121
// DLBL XIL0001,'UNIQUE.KSDS.FILE',,VSAM,CAT=IJSYSCT
// EXTENT ,FBA241,1,0,26782,200
// EXEC DRD,SIZE=200K
   RESTORE=NOP,UPSI=111,BLOCKSIZE=65496,PRINT=YES,WR=MCATPASS
   RESTORE=LOGICAL,RCAT=IJSYSCT,PARTIAL=SELECT,ASSOC=YES,FD=DELETE
   UNIQUE.KSDS.FILE
   /@
   RESTORE=NOP,TAPE=RUN
/*
/&
```

Saving, Restoring, or Copying VSAM Files

When using VSAM files, specify 'SAVE/COPY/RESTORE=VSAM/LOGICAL'. The file name must be specified for all files being Saved, Copied, or Restored. DLBL and extents must be provided (as standard labels or JCL) for each 'RFNAME/SFNAME='.

If copying, the 'RFNAME= and SFNAME=' must differ. If variable length records are used, the 'VARIABLE=YES' parameter must be used when saving. The 'VARIABLE=YES' parameter is not needed when restoring.

Note: That Dr.D considers any file in which the catalog entry record length differs from the actual to be in variable format and requires the 'VA=YES' parameter.

Associated passwords must be entered using the 'SW/RW=' parameter. Non-CI format files may be saved or restored by specifying 'VS/VR=N', but due to unknown format of the NON-CI file, the file must be restored to the same (FBA/CKD) architecture from which it is saved/copied.

Before RESTORING/COPYING, the output disk file must be defined by either Dr.D or IDCAMS. If the file is defined with the 'REUSABLE' attribute Dr.D will reset the file and restore to it. Otherwise, either IDCAMS must be used to set the file to 'EMPTY' status or the automatic definition feature must be invoked to redefine the file as 'EMPTY'.

When SAVING/COPYING/RESTORING non-keyed files, an open error message (decimal 160) will be logged by DOS-VS, unless the VSAM file type is specified through the use of the 'VS/VR=' parameter. This error can be ignored without penalty.

Keyed files may be restored as non-keyed, and non-keyed as keyed, provided they are in key sequence.

Null files (empty) may be Saved, Restored, or Copied and the null file will be initialized by writing a dummy record into the file and then deleting it. This allows the file to then be opened for any type of access (not just sequential output). The first and only record on a backup of a null file will contain 'DOCTOR VSAM NULL FILE' beginning in position 1 for fixed length records or position 5 for variable length records. This record should be taken into consideration for data tape. The data tape option allows tapes to be produced in standard fixed blocked or variable blocked format. Fixed blocked data tapes from keyed files can be input to Dr.D to produce ISAM files. More importantly, ISAM saves may be restored as VSAM files, greatly facilitating conversion efforts.

Additionally, user created data tapes may be restored by Dr.D, but must be in key sequence (if keyed) and in either standard fixed or variable format.

Example 1. Save VSAM file to tape.

This example saves VSAM file to tape.

```
// JOB V1 SAVE INDIVIDUAL VSAM FILE
// DLBL VSAMFLE,'MASTER',,VSAM
// TLBL TAPEOUT
// ASSGN SYS007,280
// EXEC DRD,SIZE=200K
   SAVE=VSAM,SFILE=VSAMFLE,SWORD=PASSWORD
/*
/&
```

Example 2. Restore VSAM file.

This example restores the file saved in Example 1.

```
// JOB V2 RESTORE INDIVIDUAL VSAM FILE
// DLBL VSAMFLE,'MASTER',,VSAM
// TLBL TAPEIN
// ASSGN SYS008,280
// EXEC DRD,SIZE=200K
   RESTORE=VSAM,RFILE=VSAMFLE,RWORD=PASSWORD
/*
/&
```

Note: The file is reusable, so 'REDEF' is not needed.

Example 3. Restore ISAM file as VSAM.

This example restores an ISAM file as a VSAM file. It uses IDCAMS to create a catalog entry before restoring.

```
// JOB V3
// DLBL OLDFILE,'ISAM FILE',,ISE
// EXTENT SYS001,000001,4,1,12,12
// EXTENT SYS002,000002,1,2,12,1200
// EXTENT SYS003,000003,1,3,12,1200
// EXTENT SYS001,000001,2,4,24,600
// TLBL TAPEOUT
// ASSGN SYS007,280
// ASSGN SYS001,DISK,VOL=000001
// ASSGN SYS002,DISK,VOL=000002
// ASSGN SYS003,DISK,VOL=000003
// EXEC DRD,SIZE=200K
   SAVE=DATA,SFILE=OLDFILE
/*
// MTC REW,SYS007
*  CREATE NEW FILE
// DLBL NEWFILE,'NEW.VSAM',,VSAM
// EXEC IDCAMS,SIZE=AUTO
   DEFINE CLUSTER (NAME(NEWFILE)-
   BUFFERSPACE(6144) FREESPACE(0 5) REUSE-
   KEYS(5 0) RECORDSIZE(80 80)-
   SPEED VOLUMES(V00001) SHAREOPTIONS(2))-
   DATA (CISZ(2048) CYL(1) FILE(DFILE) NAME(NEW.DATA))-
   INDEX (CYL(1) FILE(XFILE) NAME(NEW.INDEX))
/*
* RESTORE AS VSAM
// ASSGN SYS008,SYS007
// DLBL VSAMFLE,'NEW.VSAM',,VSAM
// TLBL TAPEIN
// EXEC DRD,SIZE=200K
   RESTORE=VSAM,RFILE=VSAMFLE,VRTYPE=K
/*
/&
```

Unique File Backup/Restore

VSAM 'UNIQUE' files can be saved/restored by either the catalog driven 'SAVE/RESTORE=LOGICAL' function or the 'SAVE/RESTORE=VSAM' function. In either case, additional DLBL and EXTENT JCL statements are required in order for the file to be defined. If the 'RESTORE= LOGICAL' function is used the generated DTF/DDNAME is altered replacing the first character portion if the file is indexed. Thus the name is changed from 'FILnnnn' to 'DILnnnn' and 'XILnnnn'. If the 'RESTORE=VSAM' function is used, the first character of the 'RFILE=' name is changed in the same way (do not start the 'RFILE=' name with 'D' or 'X').

Example 4. Unique File backup.

This example shows a UNIQUE file backup using 'SAVE=VSAM'

```
// JOB V4 VSAM SAVE OF A UNIQUE FILE USING SAVE=VSAM
// TLBL TAPEOUT
// DLBL AIL0001,'UNIQUE.KSDS.FILE',,VSAM,CAT=IJSYSCT
// EXEC DRD,SIZE=200K
   SAVE=NOP,UPSI=111,BLOCKSIZE=65496,PRINT=YES,WS=MCATPASS
   SAVE=VSAM,SCAT=IJSYSCT,SFILE=AIL0001,AUTO=YES,VUSE=SAVE,FAST=YES
   /@
   SAVE=NOP,TAPE=REW
/*
/&
```

Example 5. Unique File restore.

This example is a restore of the previously saved UNIQUE file.

```
// JOB V5 VSAM RESTORE OF A UNIQUE FILE USING 'RESTORE=VSAM'
// TLBL TAPEIN,'TAPEOUT'
// DLBL AIL0001,'UNIQUE.KSDS.FILE',,VSAM,CAT=IJSYSCT
// EXTENT ,FBA241,1,0,26982,1121
// DLBL DIL0001,'UNIQUE.KSDS.FILE',,VSAM,CAT=IJSYSCT
// EXTENT ,FBA241,1,0,26982,1121
// DLBL XIL0001,'UNIQUE.KSDS.FILE',,VSAM,CAT=IJSYSCT
// EXTENT ,FBA241,1,0,26782,200
// EXEC DRD,SIZE=200K
   RESTORE=NOP,UPSI=111,BLOCKSIZE=65496,PRINT=YES
   RESTORE=VSAM,RFILE=AIL0001,AUTO=YES,RCAT=IJSYSCT,WR=MCATPASS
   /@
   RESTORE=NOP,TAPE=RUN
/*
/&
```

Summary of Control Card Format for VSAM Data Files

Save Data File

```
SAVE=VSAM,SFILE=FFFFFFF(,SWORD=WWWWWWWW,VARIABLE=YES,DTAPE=YES,
VUSE=IGNORE/NULL/SAVE/OPERATOR/CANCEL(default),WS=MCATPASS,
VSTYPE=K/E/R/N,AUTO=YES)
(SAVE=LOGICAL)(,CPRESS=YES)(,FAST=YES)
```

Restore Data File

```
RESTORE=VSAM,RFILE=FFFFFFF(,RWORD=WWWWWWWW,DTAPE=YES,FIND=XXXXXX,
SKIP=NNN,VRTYPE=K/E/R/N,AUTO=YES/OVERRIDE,WR=MCATPASS,XL=YES/NO,
VCOMPRESS=YES)
(RESTORE=LOGICAL)
```

Copy Data File

```
COPY=VSAM,SFILE=FFFFFFF,RFILE=FFFFFFF(,VARIABLE=YES,XL=YES/NO,
VSTYPE/VRTYPE=K/E/R/N,AUTO=YES)(,SWORD=WWWWWWWW,RWORD=WWWWWWWW,
VCOMPRESS=YES)
(SAVE=VSAM,RESTORE=VSAM)
```

VSAM Auto Definition Feature

The Dr.D VSAM automatic definition feature is designed to simplify saving and restoring of 'VSAM' clusters by eliminating the need for a separate 'IDCAMS' run to delete and / or define the clusters in most circumstances.

The VSAM catalog definition for each cluster is saved by Dr.D along with the data. This definition is then used as the basis for defining the cluster during the restore. The use of this feature is optional. The default is AUTO=YES.

During the restore, Dr.D examines the file identifier in the 'DLBL' for the cluster and uses the name to override the name of the saved cluster. Using the new cluster name, Dr.D examines the target 'VSAM' catalog and selects one of the following actions.

A. If the cluster does not exist in the target catalog then Dr.D defines a new cluster using the saved definition.

B. If a cluster of the same name does exist and it contains no records, Dr.D restores into the empty cluster ignoring the saved definition.

C. If a cluster of the same name exists but is not empty, Dr.D will reuse the file if possible (even if the file is not defined as reusable). If it can't be reused, Dr.D deletes the cluster and defines a new one using the saved definition.

Note: That use of the 'FD=' parameter allows this logic to be altered so that a file is deleted and redefined, and its extents consolidated if desired. An IDCAMS printout can be obtained by use of the 'IDCAMS=PRINT' parameter.

This technique provides maximum flexibility to enable cluster definitions to be changed while at the same time providing maximum automation for regular backup and reorganization functions. It is also fully compatible with existing Dr.D job streams using 'IDCAMS'.

Device Independence

The Dr.D VSAM automatic definition feature fully supports device independence between saved and restored clusters. The catalog device type is examined during a restore and is compared with the save device type. If they are different, the space allocations for the restored cluster are adjusted in proportion to the relative track capacities (or 'FBA' blocksize) of the two device types. Full independence between 'CKD' and 'FBA' devices is supported.

Copying VSAM Clusters

The feature is fully supported during copying of a 'VSAM' cluster.

```
COPY=VSAM,SFILE=FILE001,RFILE=FILE002,AUTO=YES
```

Copying between the same or different 'VSAM' catalogs is allowed.

Multivolume VSAM Catalogs

When a VSAM catalog controls more than one volume, the automatic definition feature can restore the volumes in a number of ways.

When a 'VSAM' cluster is saved, the volume on which it resided (both prime and candidate) are recorded in its saved definition.

When restoring, these volumes are compared to the volumes owned by the catalog. If all volumes recorded for the cluster are owned by the catalog then they are used in its definition. If any of the cluster volumes are not owned by the catalog all the cluster volumes are ignored and the catalog volumes are used in the definition.

Volume allocations can be influenced in two ways:

(A) Use the IDCAMS Override Feature discussed in the following section.

(B) Use the 'RV=' parameter to direct the definition to a particular volume serial number.

```
RESTORE=VSAM,RFILE=FILEA,AUTO=YES,RVOLSER=SYSWK3,FD=D
```

IDCAMS Override Feature

Users of the VSAM automatic definition facility (AUTO=YES) can override the automatic definition on 'RESTORE=VSAM' by means of the 'IDCAMS' override feature. This feature is invoked by specifying 'AUTO=OVERRIDE' (instead of AUTO=YES) with the 'RE=VS', and then supplying IDCAMS type parameters following the 'RE=VS' statement:

```
// JOB RESTORE WITH OVERRIDE
// TLBL TAPEIN
// ASSGN SYS008,TAPE
// DLBL FILEA,'VSAM.FILE',,VSAM
// EXEC DRD,SIZE=200K
   RESTORE=NOPERATION,UPSI=111,PRINT=YES
   RESTORE=VSAM,RFILE=FILEA,AUTO=OVERRIDE,IDCAMS=PRINT
   CLUSTER(TO (99365))
   INDEX (CISZ(512))
   DATA(CISZ(4096) RECORDS(250 0))
   DATA(VOLUMES(FBA240 FBA241 FBA242 FBA243 FBA244 FBA245))
   DATA(VOLUMES(FBA251 FBA252 FBA253 FBA254 FBA255))
   /@
   RESTORE=NOP,TAPE=RUN
/*
/&
```

Each override statement must begin with either 'CLUSTER', 'DATA', or 'INDEX', and keywords and their values must be wholly contained within positions 1 to 72 of each statement. More than one keyword and value can be contained on one statement but a keyword and its value may not be split across statements. IDCAMS continuation marks are allowed but have no effect. If the 'VOLUME' parameter cannot be contained on one 72 column statement, use multiple

statements, each beginning with either 'DATA(VOLUMES(' or 'INDEX(VOLUMES('.

The override statements for each file are terminated by a '/*' before the next restore statement.

Keywords are checked for correct spelling and then passed with parameter values to IDCAMS. Unknown or incorrectly spelled keywords are ignored. Keywords are not cross checked for consistency, and values are passed to IDCAMS without verification. IDCAMS validates and reports on the resultant definition. The definition is printed along with any IDCAMS messages on SYSLST if 'IDCAMS=PRINT' is used. When deleting existing clusters, the logic changes slightly if 'AUTO=OV' is used. Normally, with 'AUTO=YES', a cluster will only be deleted if it contains records, not if it is empty. With 'AUTO=OV', the cluster is deleted even if it is empty, and a new cluster is defined using the definition from the backup together with the override specifications. Not all keywords can be specified at each level (cluster, data, index). If an override keyword is specified at a level to which it does not apply, it will be ignored. For example, if 'VOLUMES' is specified at the cluster level it will have no effect since it is only recognized at the data and index levels.

The IDCAMS override feature cannot be used at the same time as the 'RESTORE=LOGICAL' feature. However, it is still possible to restore individual files (with overrides) from a 'SAVE=LOGICAL' tape, by using 'RESTORE=VSAM' with a 'FIND=' parameter.

Dr.D LISTCAT Function

VSAM LISTCAT FUNCTION

The DR.D LISTCAT function provides a compact (but complete) report of the objects (files, paths, alternate indexes, etc.) in a specified user or master VSAM catalog. This function is almost identical to the DR.D VSAM 'SAVE LOGICAL' feature with the same file selection capability. The only difference is that no files are saved, and 'FUNCTION=LCAT' is specified instead of 'SAVE=LOGICAL'. At the end of file/object listing, a space map is printed showing how much space is allocated and available on each disk assigned to the specified catalog. Files may be selected by file ID, disk, time stamp, and file condition (see section "VSAM TUNING" on page 35, and to limit the size of the printouts see the 'FORMAT=' parameter in the keyword section).

Example 1. List all objects in a catalog.

```
// JOB DR.D LISTCAT ALL
// EXEC DRD,SIZE=200K
   FUNCTION=LCAT,SCATDDNAME=IJSYSCT
/*
/&
```

Example 2. List selected files.

```
// JOB DR.D LISTCAT SELECTIVE
// EXEC DRD,SIZE=200K
   FUNCTION=LCAT,SCATDDNAME=IJSYSCT,PARTIAL=SELECT
   PAYROLL.SYS*
   /@                        TERMINATE SELECTION
/*                           TERMINATE EXECUTION
/&
```

Example 3. List files associated with a disk.

```
// JOB DR.D LISTCAT FILES USING DISK VOLSER=CKD201
// EXEC DRD,SIZE=200K
   FUNCTION=LCAT,SCATDDNAME=IJSYSCT,SVOLID=CKD201
/*
/&
```

Example 4. List no objects, just the Space Map.

```
// JOB DR.D LISTCAT NO FILES, JUST THE SPACE MAP
// EXEC DRD,SIZE=200K
   FUNCTION=LCAT,SCATDDNAME=IJSYSCT,PARTIAL=REJECT
   *                    REJECT ALL FILES
   /@                   TERMINATE SELECTION
/*                      TERMINATE EXECUTION
/&
```

THE LISTCAT REPORT

The LISTCAT report is for the most part self-explanatory, but some fields and codes may require further explanation:

CISIZE

The first occurrence of cisize for a file is always the data cisize, while the second is always the index cisize and is indicated by an 'I' suffix.

ALLOCATIONS

Primary and secondary allocations are shown as tracks or blocks, indicated by a 'T' or 'B' suffix followed by a one digit number representing the VSAM use class.

FREE SPACE

CI (control interval) and CA (control area) are shown as whole percent. KBYTE is shown as thousands of bytes of free space available.

EXTENT INFORMATION

Extent start and size are shown as relative track or relative block, and tracks or blocks. Percent used is shown once for data and once for index regardless of the number of extents. The data extents are listed first followed by the index extents, with a relative extent number for each.

SPACE MAP AND GAP MAP

At the end of file listing for each VSAM catalog, the space owned by the catalog is displayed by disk volume, showing the extent location, size and available space. The gap map shows how the available space is distributed on the disk(s), so that space requirements can be satisfied. The gap map is also a measure of the disorganization of the space, and indicates the need for reorganization.

VSAM File Delete Function

The VSAM delete function allows files to be deleted or reset to empty status using the same flexible selection process used in the 'SAVE/RESTORE=LOGICAL' and 'LISTCAT' functions. This can be useful when reorganizing VSAM space or when moving files from one catalog to another, or anytime single/groups of files are to be deleted. Like the other VSAM catalog driven functions, file selection may be by time stamp (TSTAMP=), allowing deletion of obsolete files.

To prevent the accidental deletion of files, this function uses a slightly different approach to file selection- no file is reset or deleted unless 'PARTIAL=SELECT/REJECT' is entered. IF the 'PARTIAL' parameter is not entered, the only action taken is the listing of the files that would be deleted/emptied. Additionally, the 'FD=TEST' parameter should be used initially to produce a list of files to be deleted/reset. Once the list has been checked for correctness, then 'FD=DELETE' or 'FD=RESET' can be used to actually delete/empty the file.

Example 1. Test Run of the DELETE Function, (no files will be deleted).

The following example produces a list of files in a 'PAYROLL' system that have been moved to a new catalog (using 'SAVE=LOGICAL' and 'RESTORE=LOGICAL'), and now must be deleted from the source catalog.

```
// JOB DELETE VSAM (TEST RUN, NO FILES DELETED)
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// EXEC DRD,SIZE=200K
   FUNCTION=DELETE,RCAT=IJSYSUC,FDELETE=TEST,PARTIAL=SELECT
   PAYROLL*
/@                                      TERMINATES FILE SELECTION
/*                                      TERMINATES EXECUTION
/&
```

After checking the file list, the job is resubmitted as follows:

```
// JOB DELETE VSAM
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// EXEC DRD,SIZE=200K
   FUNCTION=DELETE,RCAT=IJSYSUC,FDELETE=DELETE,PARTIAL=SELECT
   PAYROLL*
/@                                      TERMINATES FILE SELECTION
/*                                      TERMINATES EXECUTION
/&
```

Example 2. Deleting files by their time stamp

In this example, files that have not been changed in one year are to be deleted.

```
// JOB DELETE OBSOLETE FILES
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// EXEC DRD,SIZE=200K
   FUNCTION=DELETE,RCAT=IJSYSUC,FD=DELETE,PA=SEL,TSTAMP=<-365
   *
/@                                          TERMINATES FILE SELECTION
/*                                          TERMINATES EXECUTION
/&
```

Example 3. The DELETE Function in conjunction with Space Reorganization.

In this example, a full reorganization of the space assigned to the catalog is desired to eliminate fragmentation and excessive secondary allocations. First, all the files in the catalog are saved using 'SAVE=LOGICAL' and the duplicate backup tape feature, in case of tape failure. Then the catalog is emptied (following example), and 'RESTORE=LOGICAL' is used to reload the files (using 'FDEF=CONSOLIDATE' to eliminate excessive secondary allocations).

```
// JOB DELETE ALL FILES IN VSAM CATALOG
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// EXEC DRD,SIZE=200K
   FUNCTION=DELETE,RCAT=IJSYSUC,FDELETE=DELETE,PARTIAL=SELECT
   *
/@                                          TERMINATES FILE SELECTION
/*                                          TERMINATES EXECUTION
/&
```

Note that 'PARTIAL=SELECT' is used even though all files are to be deleted, and that the '*' selector then selects all files. Also be aware that if no 'FDELETE=' is entered, that 'FDELETE=TEST' is assumed and no files or records are deleted.

An additional use of the 'FUNCTION=DELETE' feature is to empty the specified files of data records rather than delete them (use 'FDELETE=RESET' instead of 'FDELETE=DELETE'). This can be useful in limited cases, such as test files, where an empty file is desired. Although it would be nice to be able to use this approach for catalog space reorganization (the delete/define function is fairly inefficient), this is not possible at this time for several reasons: (1) the primary allocation is not released, (2) Files exceeding 16 extents can't be reset, (3) Non-CI format files cannot be reset.

VSAM TUNING

Dr.D provides features that allow VSAM files to be reorganized and/or altered to improve efficiency and/or disk utilization. This is achieved by selecting files for backup by their condition (CI/CA splits and allocations) and/or definition characteristics, and then restoring them to eliminate splits and excessive secondary allocations and/or altering catalog definitions.

During backup or listcat, additional file selector keywords are provided to select files by number of CA/CI splits, number of allocations, percent of file space in use, and CI size. These keywords must be entered in the same manner as file ID selectors, must not be abbreviated, and cannot be entered on the SAVE/RESTORE/COPY/FUNCTION control card. They also must be used in conjunction with PARTIAL=SEL/REJ, and are always used to select files, i.e. files must first be selected by the 'file condition' selectors before other selection/rejection criteria are considered.

These additional 'file condition' selectors are:

ALLOCS=NNN	SELECTS FILES BY NUMBER OF ALLOCATIONS
CISPLIT=NNN	SELECTS FILES BY NUMBER OF CI SPLITS
CASPLIT=NNN	SELECTS FILES BY NUMBER OF CA SPLITS
CISIZE=NNNNN	SELECTS FILES BY DATA CISIZE
FREESPACE>NN	SELECTS FILES BY FILE FREESPACE %

Once the files are selected and saved, additional generic/global keywords are provided to cause file definition modification and a reorganizing restore. These keywords are entered on the RESTORE= LOGICAL control card, and affect each file restored.

These additional restore keywords are:

FDEFINE=C	Causes primary allocation size recomputation
FDEFINE=M	Causes primary allocation size recomputation
FDEFINE=D	Use if FDEFINE=C is not entered and redefinition is required.
FREESPACE=CI/CA	Alters CI and CA freespace
CN=DATA/INDEX	Alters data and index CI size
BUFF=NNNN	Alters buffer space
CD=PS	Alters file primary, secondary allocation class
CX=PS	Alters index primary, secondary allocation class
SO=DATA/INDEX	Alters data and index share options

Buffer space and CI sizes are entered as either an absolute value (greater than 256) or as a multiplier of each file's buffer or CI size (BUFF=4096 sets each file's buffer to 4096 while BUFF=2 would double each file's existing buffer size).

Example 1. Save and Reorganize files with the 'CA' and 'CI' parameters.

Save and reorganize files in a catalog that have more than one CA split or more than five CI splits, or more than four allocations. The catalog has one large file spanning five disks that cannot have its allocations reduced, so reject it during the backup.

```
// JOB SAVE AND REORGANIZE SELECTED FILES
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
   SAVE=NOP,UPSI=111,PRINT=LIST,CP=YES
   SAVE=LOGICAL,SCAT=IJSYSUC,PARTIAL=REJECT,ASSOCIATES=YES
   CASPLIT=1
   CISPLIT=5
   ALLOCS=4
   VERY.BIG.VSAM.FILE
   /@
   SAVE=NOP,TAPE=REWIND
/*
// TLBL TAPEIN
// EXEC DRD,SIZE=200K
   RESTORE=NOP,UPSI=111,PRINT=LIST
   RESTORE=LOGICAL,RCAT=IJSYSUC,FDEFINE=C
   RESTORE=NOP,TAPE=RUN
/*
/&
```

Example 2. Save a number of files with a defined CI size of 2048 or smaller, and then increase the CI size on the restore.

In the same catalog, a number of files have been defined with small CI sizes, causing poor performance and poor disk track capacity utilization. Save all files with CI size 2048 or smaller, and then restore increasing CI size to 11,476.

```
// JOB SAVE AND REORGANIZE SELECTED FILES
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
   SAVE=NOP,UPSI=111,PRINT=LIST,CP=YES
   SAVE=LOGICAL,SCAT=IJSYSUC,PARTIAL=SELECT,ASSOCIATES=YES
   CISIZE=2048
   *
/@
   SAVE=NOP,TAPE=REWIND
/*
// TLBL TAPEIN
// EXEC DRD,SIZE=200K
   RESTORE=NOP,UPSI=111,PRINT=LIST
   RESTORE=LOGICAL,RCAT=IJSYSUC,FDEFINE=D,CNSIZE=11476
   RESTORE=NOP,TAPE=RUN
/*
/&
```

Example 3. Save all files in a catalog that have excessive 'CA' and 'CI' splits due to inadequate freespace. Then restore increasing the 'CA' freespace to 20% and the 'CI' freespace to 10%.

In the same catalog, some files have high activity with frequent addition of records throughout the key range of the file. The Dr.D LISTCAT report reveals that the files are having excessive CA and CI splits due to inadequate free space. Save the selected files and restore them increasing the CI freespace to 10% and the CA freespace to 20%.

```
// JOB LIST FILES FOR FREE SPACE CHANGES
// EXEC DRD,SIZE=200K
   FUNCTION=LCAT,SCAT=IJSYSUC,PARTIAL=SELECT,ASSOCIATES=YES
   CASPLIT=1
   CISPLIT=1
   *
/@
/*
/&

// JOB SAVE AND REORGANIZE SELECTED FILES
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
   SAVE=NOP,UPSI=111,PRINT=LIST,CP=YES
   SAVE=LOGICAL,SCAT=IJSYSUC,PARTIAL=SELECT,ASSOCIATES=YES
   VSAM.KSDS.FILE.ONE
   VSAM.KSDS.FILE.TWO
   VSAM.KSDS.FILE.THREE
/@
   SAVE=NOP,TAPE=REWIND
/*
// TLBL TAPEIN
// EXEC DRD,SIZE=200K
   RESTORE=NOP,UPSI=111,PRINT=LIST
   RESTORE=LOGICAL,RCAT=IJSYSUC,FDEFINE=C,FREESPACE=10/20
   RESTORE=NOP,TAPE=RUN
/*
/&
```

VSAM File VERIFY Function

The VSAM VERIFY function allows files to be VERIFIED using the same flexible selection process used in the 'SAVE/RESTORE=LOGICAL' and 'LISTCAT' functions. This can be useful after system failure to insure that all files can be opened before restarting online systems, thereby preventing further system failure.

Example 1. This example will do a selected file verify on a VSAM catalog.

```
// JOB VERIFY SELECTED FILES
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// EXEC DRD,SIZE=200K
   FUNCTION=VERIFY,SCAT=IJSYSUC,PARTIAL=SELECT
   PAYROLL*
   /@                                   TERMINATES FILE SELECTION
/*                                      TERMINATES EXECUTION
/&
```

Example 2. This example will verify all the files in a VSAM catalog.

```
// JOB VERIFY ALL FILES IN VSAM CATALOG
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// EXEC DRD,SIZE=200K
   FUNCTION=VERIFY,SCAT=IJSYSUC
/*
/&
```

VSAM Build Alternate Index Feature

The VSAM build alternate index feature allows the user to build alternate indexes automatically with Dr.D. Alternate indexes can be built while a VSAM cluster is being saved, restored, or copied. The Dr.D VSAM build alternate index feature:

- Builds up to 99 separate alternate indexes via a single pass of the base cluster
- Builds alternate indexes up to 10 times faster than the IDCAMS BLDINDEX facility.
- Builds alternate indexes for KSDS and ESDS files
- Allows cluster records for build to be selected using record selector statements.
- Allows non-unique aix keys to be made unique by the addition of a binary suffix to each key.

Building Alternate Indexes

The alternate indexes to be built by Dr.D must first be defined by IDCAMS. The Dr.D BUILD function is then used to build the alternate indexes. If the AIX is defined but not empty, Dr.D will reset it to empty status, even if it is not defined as reusable. While the build function is being performed, the base cluster may also be saved, restored, or copied, using the Dr.D parameters for those functions. Whether save, restore, or copy is performed is determined by the key words used on the BUILD statement- if 'SF' only is present, a backup is assumed; if 'RF' only is present, a restore is assumed; if both are present, a copy is assumed and the AIX is built from the output cluster. This allows a file to be reorganized at the same time AIX(s) are being created.

The BUILD function is dependent on information generated by the Dr.D auto define feature, so 'AUTO=YES' must be in effect during the BUILD (if a restore is being performed, the corresponding backup must have used 'AUTO').

Partition and execute size will vary depending on which sort program is used and whether SVA eligible phases are in the SVA. If a RESTORE/COPY method is used, more GETVIS space is needed for VSAM buffers. For best performance, use the largest partition possible and the Dr.D 'VB' and 'AB' parameters.

Along with the 'SF' and 'RF' keywords, the 'AF' keyword must be used to specify the DD/DLBL name(s) for the AIX(s) to be built. If a one or two digit number is entered, then it specifies the number of alternate indexes to be built and causes Dr.D to generate DD/DLBL names in the following format: AIXaabb where 'aa' is the number of the 'FUNCTION=BUILD' command, and 'bb' is the aix file number. The 'aa' is a sequential number that is reset at each Dr.D execution. The 'bb' is a sequential number that is reset for every 'FUNCTION=BUILD' command.

If the 'AF=' operand is not a one or two digit numeric, then it is assumed to be the actual DD/DLBL name to be used, and to represent a standard or user DLBL.

The cluster records used to build the AIX may be selected by adding 'PARTIAL=SELECT' to the FUNCTION control card and including a record selector card for each AIX to be built selectively. If multiple AIX's are built, then each selector must be prefixed by the relative AIX number (the first AIX is 1, the 2nd is 2, etc.). See section "Build AIX Record Selection" on page 46 for more about the record selection language.

Non-contiguous key AIX's may be built by specifying PARTIAL=SELECT on the FUNCTION=BUILDAIX control card and then including one or more KEY definition statements having the following format:

```
KEY=P1,L1,P2,L2,..Pn,Ln
```

The 'P' represents the relative position in the base cluster while the 'L' is the length of the key data at the 'P' position, relative to one (1,3 means position 1 length 3). Since IDCAMS and VSAM do not support non-contiguous keys directly, the AIX must be defined as 'NOUPGRADE', and the key specified need only have a length great enough to contain the total length of key segments specified. Please note that a patch to VSAM module IKQVRM is required to allow use of non-contiguous key AIX's.

AIX records may be made 'unique' by use of the 'UNIQUE' keyword. This is achieved by appending a four byte key suffix to each AIX key. The suffix is a sequential binary number that thereby makes each key unique. The defined size of the AIX key must therefore be four bytes larger to accommodate the suffix, and must be defined as 'NOUPGRADE'. If UNIQUE=A is specified and the AIX is defined as UNIQUE, then duplicate keys are dropped.

If specified on the 'FUNCTION=BUILD' control statement, 'UNIQUE' then applies to each of the AIX's created by that statement. If 'PARTIAL=SELECT' is specified, then each AIX can have its own 'UNIQUE=' keyword, allowing for instance, all to be 'UNIQUE=YES' with some being 'UNIQUE=NO' or 'UNIQUE=A' and vice versa.

Example 1. Define alternate index and associated path.

```
// JOB DEFINE ALTERNATE INDEX AND PATH
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// EXEC IDCAMS, SIZE=AUTO
   DEFINE AIX ( -
    NAME(DATA.MASTER.AIX)    -
    RELATE(DATA.MASTER.BASE) -
    BLOCKS(1000 1000)        -
    KEYS(14 7)               -
    RECORDSIZE(10 2048)      -
    NONUNIQUEKEY             -
    DATA (NAME(DATA.MASTER.AIX.D)))
   DEFINE PATH ( -
     NAME(DATA.MASTER.AIX.PATH))
/*
/&
```

Example 2. Build alternate index during save of base cluster.

This example builds a single alternate index(AIX0101) during the save of the base cluster(BASE). A sort work file(SORTWK1) is required and may need to be assigned to SYS010 (DR.D specifies SYS010 to the sort but the number of sort work files and the SYSNOs used can be varied by use of the DRD 'WORK' keyword).

```
// JOB SAVE BASE CLUSTER & BUILD AIX
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// DLBL AIX0101,'DATA.MASTER.AIX',,VSAM
// DLBL BASE,'DATA.MASTER.BASE',,VSAM
// TLBL TAPEOUT,'DATA.MASTER.BASE'
// ASSGN SYS007,280     (output tape)
// DLBL SORTWK1,,0
// EXTENT SYS001,WORK01......
// ASSGN SYS001,DISK,VOL=WORK01,SHR
// DLBL SORTWK2,,0
// EXTENT SYS002,WORK01......
// ASSGN SYS002,DISK,VOL=WORK02,SHR
// EXEC DRD,SIZE=1400K
   FUNCTION=BUILDAIX,SF=BASE,AF=1,VA=YES,AB=200,WORK=2/1
/*
/&
```

Example 3. Build alternate index during dummy save of base cluster.

This example illustrates how to build an alternate index by just reading the base cluster. The output tape is assigned to ignore to indicate to Dr.D that this a dummy save.

```
// JOB BUILD AIX BY READING BASE CLUSTER
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// DLBL AIX0101,'DATA.MASTER.AIX',,VSAM
// DLBL BASE,'DATA.MASTER.BASE',,VSAM
// ASSGN SYS007,IGN     (output tape)
// DLBL SORTWK1,,0
// EXTENT SYS010,WORK01......
// ASSGN SYS010,DISK,VOL=WORK01,SHR
// EXEC DRD,SIZE=1400K
   FUNCTION=BUILDAIX,SF=BASE,AF=1,VA=YES,AB=200
/*
/&
```

Example 4. Build alternate index during restore of base cluster.

This example builds an alternate index(AIX0101) during the restore of the base cluster(BASE).

```
// JOB RESTORE BASE CLUSTER & BUILD AIX
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// DLBL AIX0101,'DATA.MASTER.AIX',,VSAM
// DLBL BASE,'DATA.MASTER.BASE',,VSAM
// TLBL TAPEIN,'DATA.MASTER.BASE'
// ASSGN SYS008,280    (input tape)
// DLBL SORTWK1,,0
// EXTENT SYS010,WORK01.....
// ASSGN SYS010,DISK,VOL=WORK01,SHR
// EXEC DRD,SIZE=1400K
   FUNCTION=BUILDAIX,RF=BASE,AF=1,VA=YES,AB=500
/*
/&
```

Example 5. Build alternate index during copy of base cluster.

This example builds an alternate index(AIX0101) during the copy of the base cluster(BASEIN).

```
// JOB COPY BASE CLUSTER & BUILD MULTIPLE AIX'S
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// DLBL AIX0101,'DATA.MASTER.AIX',,VSAM
// DLBL BASEIN,'DATA.MASTER.BASE',,VSAM
// DLBL BASEOUT,'DATA.MASTER.BASE.SAVE',,VSAM
// DLBL SORTWK1,,0
// EXTENT SYS010,WORK01........
// ASSGN SYS010,DISK,VOL=WORK01,SHR
// EXEC DRD,SIZE=1400K
   FUNCTION=BUILDAIX,SF=BASEIN,RF=BASEOUT,AF=1,VA=YES,AB=500
/*
/&
```

Example 6. Build multiple alternate indexes during restore.

The number of alternate indexes to be built is specified via the 'AF=' keyword ('AF=3').

```
// JOB RESTORE AND BUILD MULTIPLE AIX'S
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// DLBL AIX0101,'DATA.MASTER.AIX.ONE',,VSAM
// DLBL AIX0102,'DATA.MASTER.AIX.TWO',,VSAM
// DLBL AIX0103,'DATA.MASTER.AIX.THREE',,VSAM
// DLBL BASE,'DATA.MASTER.BASE',,VSAM
// TLBL TAPEIN,'DATA.MASTER.BASE'
// ASSGN SYS008,280   (input tape)
// DLBL SORTWK1,,0
// EXTENT SYS010,WORK01......
// ASSGN,DISK,VOL=WORK01,SHR
// EXEC DRD,SIZE=1400
   FUNCTION=BUILDAIX,RF=BASE,AF=3,VA=YES,AB=500
/*
/&
```

Example 7. Build multiple alternate indexes via multiple builds.

In this example AIX0101 and AIX0102 are built from the save of VSAM base cluster BASE1. AIX0201 and AIX0202 are built from the save of VSAM base cluster BASE2.

```
// JOB SAVE TWO BASE CLUSTERS AND BUILD MULTIPLE AIX'S
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// DLBL AIX0101,'DATA.MASTER.AIX.ONE',,VSAM
// DLBL AIX0102,'DATA.MASTER.AIX.TWO',,VSAM
// DLBL AIX0201,'EMP.MASTER.AIX.ONE',,VSAM
// DLBL AIX0202,'EMP.MASTER.AIX.TWO',,VSAM
// DLBL BASE1,'DATA.MASTER.BASE',,VSAM
// DLBL BASE2,'EMP.MASTER.BASE',,VSAM
// TLBL TAPEOUT,'MASTER.BACKUP'
// ASSGN SYS007,280    (output tape)
// DLBL SORTWK1,,0
// EXTENT SYS010,WORK01.....
// ASSGN SYS010,DISK,VOL=WORK01,SHR
// EXEC DRD,SIZE=1400K
   FUNCTION=BUILDAIX,SF=BASE1,AF=2,VA=YES,AB=200
   FUNCTION=BUILDAIX,SF=BASE2,AF=2,VA=YES,AB=200
/*
/&
```

Example 8. Build multiple alternate indexes via multiple builds and using record selection.

In this example AIX0101 and AIX0102 are built from the save of VSAM base cluster BASE1. AIX0201 and AIX0202 are built from the save of VSAM base cluster BASE2.

```
// JOB SAVE TWO BASE CLUSTERS AND BUILD MULTIPLE AIX'S
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// DLBL AIX0101,'DATA.MASTER.AIX.ONE',,VSAM
// DLBL AIX0102,'DATA.MASTER.AIX.TWO',,VSAM
// DLBL AIX0201,'EMP.MASTER.AIX.ONE',,VSAM
// DLBL AIX0202,'EMP.MASTER.AIX.TWO',,VSAM
// DLBL BASE1,'DATA.MASTER.BASE',,VSAM
// DLBL BASE2,'EMP.MASTER.BASE',,VSAM
// TLBL TAPEOUT,'MASTER.BACKUP'
// ASSGN SYS007,280    (output tape)
// DLBL SORTWK1,,0
// EXTENT SYS010,WORK01.....
// ASSGN SYS010,DISK,VOL=WORK01,SHR
// EXEC DRD,SIZE=1400K
   FUNCTION=BUILDAIX,SF=BASE1,AF=2,VA=YES,AB=200,PA=SE
   01 SELECT C2 = '    '
   02 SELECT C2 <  00400
   /@
   FUNCTION=BUILDAIX,SF=BASE2,AF=2,VA=YES,AB=200,UNIQUE=YES,PA=SE
   02 SELECT C2 = A
   /@
/*
/&
```

Example 9. Build multiple alternate indexes via multiple builds and with non-contiguous keys. Use the 'UNIQUE' keyword to cause each key to be made unique by adding a key suffix.

In this example AIX0101 and AIX0102 are built from the save of VSAM base cluster BASE1. AIX0201 and AIX0202 are built from the save of VSAM base cluster BASE2.

```
// JOB SAVE TWO BASE CLUSTERS AND BUILD MULTIPLE AIX'S
// DLBL IJSYSUC,'VSAM.USER.CATALOG',,VSAM
// DLBL AIX0101,'DATA.MASTER.AIX.ONE',,VSAM
// DLBL AIX0102,'DATA.MASTER.AIX.TWO',,VSAM
// DLBL AIX0201,'EMP.MASTER.AIX.ONE',,VSAM
// DLBL AIX0202,'EMP.MASTER.AIX.TWO',,VSAM
// DLBL BASE1,'DATA.MASTER.BASE',,VSAM
// DLBL BASE2,'EMP.MASTER.BASE',,VSAM
// TLBL TAPEOUT,'MASTER.BACKUP'
// ASSGN SYS007,280    (output tape)
// DLBL SORTWK1,,0
// EXTENT SYS010,WORK01.....
// ASSGN SYS010,DISK,VOL=WORK01,SHR
// EXEC DRD,SIZE=1400K
   FUNCTION=BUILDAIX,SF=BASE1,AF=2,VA=YES,AB=200,PA=SE
   01 KEY=1,4,100,10,150,5
   02 UNIQUE=YES
   /@
   FUNCTION=BUILDAIX,SF=BASE2,AF=2,VA=YES,AB=200,UNIQUE=YES,PA=SE
   02 KEY=10,2,1,3,25,6
   02 UNIQUE=NO
   /@
/*
/&
```

Build AIX Record Selection

Starting with DR.D version 6.2.6, selection of base cluster records for the build AIX function has been improved to allow selection based on fields not included in the AIX data fields. Selection takes place before the AIX records are passed to the sort, greatly improving the efficiency of the sort process both in processing time and sort work area requirements.

Record selection is specified by use of the DRD SELECT statement rather than the sort INCLUDE/OMIT statements. One SELECT statement is allowed for each AIX to be built and has the following format:

Bytes 0-1	A two digit number indicating the AIX being selected. IF blank, 01 is assumed.
Bytes 2-7	SELECT (or just SEL)
Bytes n-80	The record selection formula.

The record selection formula consists of one or more field selectors, each made up of three sub-fields: the field locator, a logical operator, and a constant for comparison. The field locator consists of a one byte type field allowing character

(C), hexadecimal (X), or packed decimal (P). Following the type field is a number specifying the starting position in the record (relative to one). Some examples are:

P12 - specifies a packed decimal field starting at position 12
C5 - specifies a character field starting at position 5
X37 - specifies a hexadecimal field starting at position 37

Logical Operators

The logical operator specifies the type of comparison to be made between the record field and a specified constant. It may be the '=' sign for comparing equal, the '<' for less than, the '>' for greater than, and 'NE' for not equal.

Following the logical operator is the constant data to be compared to the record field. This constant must be of an appropriate format to match the type field specified by the field locator. If packed decimal is specified, the length of the record field is determined by scanning for the sign in the rightmost position. If a character field includes blanks, enclose the constant in quotes ('). Hexadecimal fields must have an even number of proper hexadecimal digits (0-F).

The following are examples of the SELECT statement:

1. Select records containing 'FLORIDA' starting in position 45:

```
SELECT C45 = FLORIDA
```

2. Select records having a value greater than 512 in the packed decimal field starting in position 12:

```
SELECT P12 > 512
```

3. Select records having a value less than x'0000B800' in the field starting in position 138:

```
SELECT X138 < 0000B800
```

4. Select records that do not have 'DEAD' in position 1000:

```
SELECT C1000 NE DEAD
```

Complex Record Selection

Complex record selection can be specified by using the logical connectors 'V' (or) and '&' (and), and if necessary, parentheses to insure correct evaluation of the expression. Some examples are:

1. Select records having 'FLORIDA' in position 45 and a value greater than 512 in position 12:

```
SELECT C45 = FLORIDA & P12 > 512
```

2. Select records having either 'FLORIDA' or 'GEORGIA' in position 45:

```
SELECT C45 = FLORIDA V C45 = GEORGIA
```

3. Select records having 'A' in positions 5 and 10, or where the packed decimal field in position 20 is in the range 100 through 499:

```
SELECT (C5 = A & C10 = A) V (P20 > 99 & P20 < 500)
```

Note that spaces must separate field locators, logical operators, constants and logical connectors, but spaces are not required after the '(' or before the ')'.

Volume Functions

Saving, Restoring, Copying Volumes

Dr.D SAVES, RESTORES, or COPIES the entire or just active portions of a disk. It can also create IPL records on the save tape so the tape may later be IPL'd and restored in a standalone fashion or under VM. Two volume backup/restore/copy methods are provided, 'ALL' and 'CURRENT'. The 'ALL' method saves every block/track on the disk, while the 'CURRENT' method is 'VTOC driven' saving only the allocated space on the disk, the IPL track/block and the VTOC.

For speed and flexibility, the 'CURRENT' method should be used in most cases, while the 'ALL' method should be reserved for disks lacking a VSE type VTOC (the VM sysres for example).

At this time, functionality is more limited for the Fixed Block Architecture (FBA) disk, while functionality for Count Key Data (CKD) disk is more complete. If an example has 'CKD DISK ONLY' or 'FBA DISK ONLY' indicated, you must be sure that your disk is of the correct type. If you are not sure of the disk type, note that FBA disk space is allocated in blocks and uses relative block addressing while CKD disk are allocated in tracks/cylinders and use cylinder-head-record or relative track addressing.

For RESTORE=ALL/CURRENT, backup & restore disk must be same type. Only SAVE=ALL/CURRENT/PHYSICAL backups can be restored standalone.

Saving disk volumes to tape.

Dr.D will save an entire volume if 'SAVE=ALL' is specified, but the 'SHALT' parameter can be used to set the upper/ending disk address to be saved.

Dr.D will save any unexpired file extents, IPL track, and VTOC if 'SAVE=CURRENT' is specified. Expired files may be included or excluded by use of the 'EXP=YES/NO/+NNN/YYDDD' parameter. If 'YES' is specified, expired files are saved. If the assumed value 'NO' is used, only those files that are unexpired as of the IPL date are saved. If '+NNN' is specified, the 'NNN' is added to the IPL date for expiration checking. If 'YYDDD' is specified, it is used for expiration checking instead of the IPL date.

If neither 'SVOLSER' nor 'SSYSNO' is specified then the disk having the assumed SYSNO assigned to it is used as input to backup or copy. The assumed SYSNO as supplied is SYS001 but see "Appendix II - Setting Assumed Options" on page 199 for more information.

To save a volume or disk area for standalone restoration:

'SAVE=IPL' followed by 'SAVE=ALL/CURRENT/PHYSICAL'.

Example 1. SAVE=CURRENT VTOC driven backup disk to tape.

This job stream saves the SYSWK1 disk volume by the SAVE=CURRENT method. Expired files are saved, the tape is compressed, and the saving of each sequential (SD) file ceases on reading the end-of-file (EOF) after the last data block.

```
                      // JOB EXAMPLE 1 SAVE=CURRENT DISK BACKUP
                      // ASSGN SYS007,280
                      // MTC REW,SYS007
                      // TLBL TAPEOUT,'MY PACK SAVE'
                      // EXEC DRD,SIZE=200K
set global options->     SAVE=NOP,UPSI=111,PRINT=LIST,CP=YES
save the disk------>     SAVE=CURRENT,SV=SYSWK1,EXPIRED=YES,EOF=YES
unload the tape---->     SA=NOP,TAPE=RUN
                      /*
                      /&
```

Example 2. Save ALL tracks/blocks of disk to tape

This job stream saves a VM sysres that occupies the first 100 cylinders of a 3380. The disk is defined as a full/native 3380 to VSE.

```
                      // JOB EXAMPLE 2 SAVE=ALL VOLUME BACKUP
                      // ASSGN SYS007,280
                      // MTC REW,SYS007
                      // TLBL TAPEOUT,'MY PACK SAVE'
                      // EXEC DRD,SIZE=200K
set global options->     SAVE=NOP,UPSI=111,PRINT=LIST,CP=YES
save the disk------>     SAVE=ALL,SV=VMSRES,SHALT=9914
unload the tape---->     SA=NOP,TAPE=RUN
                      /*
                      /&
```

Example 3. Selective VTOC driven backup. (CKD DISK ONLY)

Save a 3380 disk, VOLSER CKD801, excluding VSAM data spaces and any files in cylinders 0-10. Also exclude any files whose ID's are prefixed with 'SORT' or contain the word 'TEMP'

```
                        // JOB EXAMPLE 3 SAVE=CURRENT VOLUME BACKUP
                        // ASSGN SYS007,280
                        // MTC REW,SYS007
                        // TLBL TAPEOUT
                        // EXEC DRD,SIZE=200K
set global options->       SAVE=NOP,UPSI=111,PRINT=LIST,CP=YES
save the disk------>       SAVE=CUR,SV=CKD801,EX=YES,VSAM=NO,PART=REJECT
limit disk area---->       1100-999999
reject SORT* files->       SORT*
reject TEMP's------>       =S=TEMP
end of selectors--->       /@
unload the tape---->       SA=NOP,TAPE=RUN
                        /*
                        /&
```

Example 4. Save DOSRES for standalone restore.

This example assumes that a tape manager is used, and so uses two TLBL statements. This is necessary due to the fact that DR.D closes the tape after the SAVE=IPL is completed and re-opens it for the actual backup. The format of the TLBL should be set to conform to the tape manager being used. Note that the TD= is used to point the backup to the correct TLBL.

```
                         // JOB EXAMPLE 4A STANDALONE WITH TAPE MANAGER
first file on tape--->   // TLBL TAPEOUT,'DRD.STANDALONE',,,,1
2nd file on tape----->   // TLBL BACKUP,'DOSRES.BACKUP',,,,2
                         // EXEC DRD,SIZE=200K
DRD programs to tape->      SAVE=IPL
set global options--->      SAVE=NOP,UPSI=111,PRINT=LIST,CP=YES
save the disk-------->      SAVE=CUR,SV=DOSRES,EX=YES,TD=BACKUP
                         /*
                         /&

                         // JOB EXAMPLE 4B STANDALONE WITHOUT TAPE MANAGER
                         // ASSGN SYS007,280
                         // MTC REW,SYS007
                         // TLBL TAPEOUT
                         // EXEC DRD,SIZE=200K
DRD programs to tape->      SAVE=IPL
set global options--->      SAVE=NOP,UPSI=111,PRINT=LIST,CP=YES
save the disk-------->      SAVE=CUR,SV=DOSRES,EX=YES
unload the tape------>      SA=NOP,TAPE=RUN
                         /*
                         /&
```

Restoring disk volumes from tape.

A disk may be restored to a disk whose size and location of VTOC differs from the save disk. However, at the conclusion of the RESTORE the restore disk's volume label, size and location of the VTOC will be identical to the save disk. If a 'CURRENT' function is done instead of an 'ALL' function, the VTOC of the restore disk will contain all of the labels from the saved disk even though all files weren't restored. Unwanted files can be removed from the VTOC using a DR.D 'FUNCTION=DELETE'.

Use the 'VOLOUT/VOLSER=' (VO=XXXXXX) keyword to set the serial the volume being restored to a value different from the volume number of the saved disk.

If neither 'RVOLSER=' nor 'RSYSNO=' is specified then the disk having the assumed SYSNO assigned to it is used as the output device. The assumed SYSNO as supplied is SYS002 but see "Appendix II - Setting Assumed Options" on page 199. Use of 'RVOLSER=' is recommended to avoid the possibility of restoring to the wrong disk.

Volumes saved to be restored standalone may also be restored non-standalone by either forward spacing the tape three tape marks (two if LABEL=NO was used during backup), and/or using a TLBL for the tape specifying file number two.

Example 5. Restore tape to disk.

This example restores a SAVE=ALL method backup tape to a disk having VOLSER CKD1C2. Note that the VOLSER of the target disk will be the same as the saved disk, when the restore is completed. (Use 'VOLOUT=' to either retain the target volser or set a new one.)

```
                        // JOB EXAMPLE 5 RESTORE SAVE=ALL VOLUME
                        // ASSGN SYS008,280
                        // MTC REW,SYS008
                        // TLBL TAPEIN
                        // EXEC DRD,SIZE=200K
set globals--------->      RESTORE=NOP,UPSI=011,PRINT=YES
restore whole disk-->      RESTORE=ALL,RVOLSER=CKD1C2
unload the tape----->      RESTORE=NO,TA=RU
                        /*
                        /&
```

Example 6. Restore tape to disk.

This example restores a disk volume from an IPLable backup tape. The 'VOLSER/VOLOUT=' of the output disk is set to 'OLDRES'.

```
                       // JOB EXAMPLE 6
                       // ASSGN SYS008,280
                       // MTC REW,SYS008
skip ipl part------->  // MTC FSF,SYS008,3
                       // TLBL TAPEIN,'SYSRES SAVE',,,,2
assign output disk-->  // ASSGN SYS002,137
                       // EXEC DRD,SIZE=200K
restore the disk---->     RESTORE=CURRENT,UPSI=011,VOLOUT=OLDRES
unload the tape----->     RE=NO,TA=RU
                       /*
                       /&
```

Example 7. Restore Mini Disks.

This example uses a tape created by the SAVE=ALL method. The tape contains a number of VM mini disks and will be used to selectively restore mini disks 1 , 2 and 3. The 3rd disk is to be relocated to a new physical location.

```
                       // JOB EXAMPLE 7
                       // ASSGN SYS008,280
                       // MTC REW,SYS008
                       // TLBL TAPEIN
                       // EXEC DRD,SIZE=200K
set global options->      REST=NOP,UPSI=111,PRINT=LIST
restore the disk--->      REST=ALL,RV=VMSRES,AREA=SELECT
cyl 100 thru 130--->      10000-13014
cyl 200 thru 249--->      20000-24914
cyl 300 to cyl 400->      30000-30014  TO=K40000
end selection------>      /@
unload the tape---->      RE=NOP,TAPE=RUN
                       /*
                       /&
```

Note: That the 3rd mini disk that was saved from cylinder 300 is restored to cylinder 400 and that the virtual cylinder number is retained (Kept). Whenever using the 'AREA=SELECT/REJECT' parameter that comments must not be added to the area selector card(s).

Example 8. Individual file restore. (FBA DISK ONLY)

This example performs an individual file restore from an FBA disk 'SAVE=CURRENT' BACKUP.

```
                       // JOB EXAMPLE 8 RESTORE & RELOCATE FROM SA=CURRENT
                       // TLBL TAPEIN
                       // ASSGN SYS008,280
                       // MTC REW,SYS008
DLBL for file------->  // DLBL SDFILE,'NEW VERSION OF FILE'
FBA extent---------->  // EXTENT SYS002,,1,0,557000,500
assign an FBA disk-->  // ASSGN SYS002,223
                       // EXEC DRD,SIZE=200K
set globals--------->     RESTORE=NOP,UPSI=011
restore the file---->     REST=CUR,RFILENAME=SDFILE,SIDENT=SAVED.FILE.ID
unload the tape----->     RESTORE=NOPERATION,TAPE=RUN
                       /*
                       /&
```

Individual files may be restored from the backup produced by the Volume 'SAVE=CURRENT' method by specifying the file ID of the saved file (SIDENT=). The file may be relocated to a new location by use of the 'RF=' parameter, but even if restoring to the original location, the 'RFILENAME=' is required to avoid the possibility of restoring to the wrong area.

***WARNING* IF 'RF' AND 'SI' ARE NOT ENTERED, A VOLUME RESTORE OCCURS.**

Example 9. Individual file restore from 'SAVE=CURRENT' backup. (CKD DISK ONLY)

Individual files may be restored from the backup produced by the volume 'SAVE=CURRENT' method by specifying the 'PARTIAL=' parameter and the ID(s) of the file(s) to be restored.

```
                       // JOB EXAMPLE 9 RESTORE & RELOCATE FILE
                       // TLBL TAPEIN
                       // ASSGN SYS008,280
                       // MTC REW,SYS008
                       // DLBL SDFILE1,'NEW.VERSION.OF.FILE1'
                       // EXTENT SYS002,,1,0,15,500
                       // DLBL SDFILE2,'NEW.VERSION.OF.FILE2'
                       // EXTENT SYS003,,1,0,15,500
                       // ASSGN SYS002,223
                       // ASSGN SYS003,224
                       // EXEC DRD,SIZE=200K
set globals--------->     RESTORE=NOP,UPSI=011,PRINT=YES
restore the files--->     RESTORE=CURRENT,PARTIAL=SELECT
select file 1------->     OLD.VERSION.OF.FILE1      @SDFILE1@
select file 2------->     OLD.VERSION.OF.FILE2      @SDFILE2@
end selection------->     /@
unload the tape----->     RESTORE=NOP,TAPE=RUN
                       /*
                       /&
```

Note: The DLBL name is coded on the file selector within the "@" signs, and that SYS002 must be assigned to the first disk being restored to.

***WARNING* IF 'PARTIAL=SELECT' IS NOT ENTERED, A VOLUME RESTORE OCCURS.**

Starting with DR.D version 6.1.9, files save from CKD disk using the SAVE=CURRENT method may also be restored by the RESTORE=SD or the RESTORE=DATA method. The backup must have been produced by V619 or later.

```
                      // JOB EXAMPLE 9A RESTORE FILES FROM SAVE=CURRENT
                      // TLBL TAPEIN
                      // ASSGN SYS008,280
                      // MTC REW,SYS008
                      // DLBL SDFILE1,'NEW.VERSION.OF.FILE1.'
                      // EXTENT SYS002,,1,0,15,500
                      // DLBL SDFILE2,'NEW.VERSION.OF.FILE2'
                      // EXTENT SYS003,,1,0,15,500
                      // ASSGN SYS002,223
                      // ASSGN SYS003,224
                      // EXEC DRD,SIZE=200K
set globals--------->    RESTORE=NOP,UPSI=111,PRINT=YES
restore thru liocs-->    RESTORE=SD,RF=SDFILE1,FI='OLD.VERSION.OF.FILE1'
restore thru piocs-->    RESTORE=DA,RF=SDFILE2,FI='OLD.VERSION.OF.FILE2'
unload the tape----->    RESTORE=NOP,TAPE=RUN
                      /*
                      /&
```

Compacting/Reorganizing disk or pools.

Reorganization/compacting of disk manager pool space on CKD disk can be achieved with a minimal impact on processing, using the RESTORE=CURRENT,AREA=COMPACT method. This method can be used with the three major disk management systems- CA-DYNAM, EPIC, and CA-SYSTEM-MANAGER. This method is preferable to the RESTORE=LOGICAL method for a number of reasons:

1. It is substantially faster.
2. Because the files are not opened, critical information in the VTOC such as generation numbers, creation dates, etc. are not changed.
3. One disk at a time can be compacted even though there may be files on the disk that have extents on other disks, allowing easier management of the compacting operation.

For this method to operate correctly, several aspects must be considered:

1. At this time, this method can only be used with the CKD disk type.
2. During SAVE or RESTORE of the disk being compacted, no files on that disk can be allocated or deleted, and none accessed during the RESTORE compacting operation. The HV= keyword can be used to prevent problems

in this area, or the disk can be detached from the disk manager while compacting.

3. If a VTOC index is used, it must be regenerated after compacting is completed.
4. The System Manager Verify function must be run against the compacted disk to update the System Manager extent information.
5. If EPIC cataloged files are used, the job should be set up so that DR.D can update the EPIC catalog (see example).

Example 10. Disk reorganization using 'RESTORE=CURRENT'. (CKD DISK ONLY)

This example reorganizes cylinders 1-300 and 600-849 of a 3380 disk previously saved by 'SAVE=CURRENT'. No cylinder boundary alignment is needed.

```
                       // JOB EXAMPLE 10 REORGANIZE SELECTED AREAS
                       // TLBL TAPEIN
                       // ASSGN SYS008,280
                       // EXEC DRD,SIZE=200K
set global options-->     RESTORE=NOPERATION,UPSI=111,PRINT=YES
restore the disk---->     REST=CURRENT,RV=MYVSER,AREA=COMPACT,ALIGN=NO
cyl 1-300---------->      100-30014
cyl 600-849-------->      60000-84914
end area specs------>     /@
unload the tape----->     RESTORE=NOP,TAPE=RUN
                       /*
                       /&
```

Example 11. Compacting EPIC pool space. (CKD DISK ONLY)

EPIC pool space can be compacted by DRD, one disk at a time using the 'SAVE=CURRENT' volume backup and restore as follows:

```
                       // JOB EXAMPLE 11 COMPACT EPIC POOL SPACE
                       // TLBL TAPEOUT
                       // TLBL TAPEIN
                       // ASSGN SYS007,XXX
                       // MTC REW,SYS007
                       // ASSGN SYS008,SYS007
DLBL for EPIC cat--->  // DLBL IJSYSDS,'EPIC.DSN'
                       // EXTENT SYS003
assgn for EPIC cat-->  // ASSGN SYS003,YYY
                       // EXEC DRD,SIZE=200K
set global options-->     SA=NO,UP=111,CP=YES
save the disk------->     SA=CU,SV=DISK01,EX=YES,HV=YES
rewind the tape----->     SA=NO,TA=RE
restore & compact--->     RE=CU,RV=DISK01,DM=EPIC,ALI=NO,HV=HOLD,AR=COMPACT
pool area cyl 1-883->     100-88314
end area specs------>     /@
unload the tape----->     RE=NO,TAPE=RUN
                       /*
                       /&
```

The DLBL for the EPIC catalog must be present here or in standard labels. Its extent must be assigned to the correct disk.

Specify the area of the disk to be compacted in cylinder and head notation. If the catalog is on this disk, exclude it from the area.

DR.D will lock the disk while the backup and restore is taking place as indicated by use of the 'HV=' (HOLD VTOC) keyword, and will lock the EPIC catalog while the catalog is being updated. When first using this feature, you should stop system activity that would cause the EPIC catalog to be updated, get a backup of the catalog, and thoroughly check the catalog after the compacting to insure that the catalog and VTOC of the compacted disk are in agreement. Then if any problems are evident, the catalog can be restored, and the compacted disk can be restored without 'AREA=COMPACT'.

Copying volumes disk to disk.

A disk may be copied to a disk whose VTOC is of a different size and location from the save disk. However, at the conclusion of the COPY the restore disk's volume label, size and location of the VTOC will be identical to the save disk. If a 'CURRENT' function is done instead of an 'ALL' function, the VTOC of the restore disk will contain all of the labels from the saved disk even though all weren't copied. If you do not want those labels in the VTOC, use the DR.D 'FUNCTION=DELETE' to remove them.

Use the 'VOLSER/VOLOUT=' (VO=XXXXXX) keyword to set the serial number of the output disk to a value different from the volume serial number of the input disk.

If neither 'RVOLSER=' nor 'RSYSNO=' is specified then the disk having the assumed SYSNO assigned to it is used as the output device. The assumed SYSNO as supplied is SYS002 but see "Appendix II - Setting Assumed Options" on page 199. Use of 'RVOLSER=' is recommended to avoid the possibility of copying to the wrong disk.

Example 12. Copy Current areas disk to disk.

This example copies the active areas of a disk (VOLSER 111111) to another disk (VOLSER 222222), retaining the volser of the target disk (VOLOUT=222222). Expired files are not copied- add 'EXPIRED=YES' to include expired files.

```
                        // JOB EXAMPLE 12 COPY DISK TO DISK
                        // EXEC DRD,SIZE=200K
set globals--------->      COPY=NOP,UPSI=011,PRINT=YES
copy the disk------->      COPY=CUR,SVOL=111111,RVOL=222222,VOLOUT=222222
                        /*
                        /&
```

Example 13. Copy all blocks/tracks from one disk to another.

This example copies a 3380 from 151 to a 3380 at 152.

```
                        // JOB EXAMPLE 13 COPY ALL TRKS/BLKS DISK TO DISK
                        // ASSGN SYS001,151
                        // ASSGN SYS002,152
                        // EXEC DRD,SIZE=200K
set globals--------->      COPY=NOP,UPSI=011,PRINT=YES
copy the disk------->      COPY=ALL,VO=NEWVOL
                        /*
                        /&
```

Summary of Control Card Format for Dr.D Volume Operations

IPLable Tape

Creates an IPLable tape for standalone volume restore.

```
SAVE=IPL(,LABEL=NO)
```

Save Entire Disk

Saves entire disk.

```
SAVE=ALL(,SSYSNO=SYSXXX)(,BLOCKSIZE=NNNNN)(,CPRESS=YES)(,SHALT=X)
```

Save Active Areas

Saves active areas of a disk.

```
SAVE=CURRENT(,SSYSNO=SYSXXX)(,EOF=YES)(,EXPIRED=YES/NO/+NNN/YYDDD)
(,BLOCKSIZE=NNNNN)(,CPRESS=YES)
```

Restore from SAVE=ALL backup

```
RESTORE=ALL(,RSYSNO=SYSXXX)/(,RVOLSER=VVVVVV)(,AREA=SELECT/REJECT)
(,FIND=(XXXXXX))(,TI=SYSXXX)(,BLOCKSIZE=NNNNN)
(,VOLSER/VOLOUT=VVVVVV)
```

Restore from SAVE=CURRENT backup

```
RESTORE=CURRENT(,RSYSNO=SYSXXX)/(,RVOLSER=VVVVVV)(,TI=SYSXXX)
(,AREA=SELECT/REJECT/COMPACT)(,ALIGN=NO/YES/ALL)(,RFILE=XXXXXXX)
(,FIND=(XXXXXX))(,SIDENT=II..II)(,BLOCKSIZE=NNNNN)
(,VOLSER/VOLOUT=VVVVVV)
```

Copy Entire Disk

```
COPY=ALL(,SSYSNO=SYSXXX)(,RSYSNO=SYSXXX)(,SHALT=X)
(,SVOLSER=VVVVVV)(,RVOLSER=XXXXXX)(,VOLSER/VOLOUT=VVVVVV)
```

Copy Active Areas

```
COPY=CUR(,SSYSNO=SYSXXX)(,RSYSNO=SYSXXX)(,EOF=YES
(,SVOLSER=VVVVVV)(,RVOLSER=XXXXXX)(,VOLSER/VOLOUT=VVVVVV)
(,EXPIRED=YES/NO/+NNN/YYDDD)
```

Default Values for Volumes (See appendix II)

```
SSYSNO=SYS001
RSYSNO=SYS002
```

Physical SAVE/RESTORE/COPY

Saving, Restoring, Copying Physical Operations

Dr.D enables users to SAVE/COPY/RESTORE portions of a disk on a physical track/block basis, independent of Volumes, Data Files, or Libraries.

This function is valuable for its ability to manipulate data at the physical level. This poses some risk in terms of security and possible inadvertent damage to critical data. For that reason, it may be wise to limit its use to qualified authorized personnel. This can be achieved by limiting access to one DR.D module- DOCTOR02. If a security package is available, the use of DOCTOR02 can be easily controlled, otherwise DOCTOR02 can be placed in a restricted library, or kept on tape and only restored by authorized persons.

This function offers some useful benefits. For example, consider a disk where the third track in the VTOC has been corrupted. All of the VTOC labels are in the first two tracks, but even though the files are accessible, no new files can be added. If we could copy track 4 to track 3, then the VTOC would be fixed and usable.

Using Dr.D on a physical track basis, the VTOC can be repaired quickly without unloading/reloading the disk's files. Specifying 'SAVE/COPY/RESTORE=PH' (PHYSICAL) will cause Dr.D to operate in a track/block physical mode. The save disk is assumed to be assigned to SYS001, but can be overridden by setting 'SSYSNO=SYSxxx'. The restore disk is assumed to be assigned to SYS002 but may be overridden by setting 'RSYSNO='. (They may both be replaced by using 'CSYSNO='.) Anytime the 'SV=' or 'RV=' volsers are entered, assignments are not needed provided that an assignment already exists or a temporary assignment can be made by Dr.D.

Dr.D starts by using the 'SBEGIN=' and 'RBEGIN=' operands (for saving and restoring). The 'SHALT=' and 'RHALT=' operands dictate when the saving and restoring cease. The disk addresses entered are one to eight numeric digits in length, with the suffix 'Rnnn' if record (block) level processing is invoked by 'RMODE=A/U'.

If the target disk is a CKD type, the two digits at the extreme right are assumed to be the head number, with the rest of the number as the cylinder number. If the target disk is an FBA type, the whole number is the FBA block number. In any case, leading zeros are not required. If an ending address is not entered, it is made equal to the start address and a single track/FBA block operation is performed. If copying and the input and output disk addresses are the same they may be entered just once using the 'CBEGIN=' and 'CHALT=' operands.

Whenever you are copying or restoring in the physical sense (CO/RE=PH), you must specify the Restore Mode parameter 'RM=', if the assumed value is unsatisfactory (Image mode is assumed).

Setting 'RM=S' (Squeeze) will cause Dr.D to restore the extent putting the maximum number of records it can fit on each track. Consider the following: A user saves three tracks of a 3380 that looked like this:

TRACK 1 15000-BYTE REC, 15000-BYTE REC
TRACK 2 NO RECORDS - ERASED TRACK
TRACK 3 15000-BYTE REC, 5000-BYTE REC

Copying or Restoring those three tracks with 'RM=I' would yield the exact same layout:

TRACK 1 15000-BYTE REC, 15000-BYTE REC
TRACK 2 NO RECORDS - ERASED TRACK
TRACK 3 15000-BYTE REC, 5000-BYTE REC

However, if the same operation is done with 'RM=S', the Copied/Restored tracks would look like:

TRACK 1 15000-BYTE REC, 15000-BYTE REC, 15000-BYTE REC
TRACK 2 5000-BYTE REC
TRACK 3 UNTOUCHED

Two additional values may be entered for the RMODE ('A' and 'U'). Setting 'RM=A' (Addon) will cause Dr.D to restore/copy the input blocks adding them at the CCHHR address specified by the 'RBEGIN=' parameter, in 'Squeeze' mode. If 'RM=U' is specified, then the contents of each input block is 'UPDATED' into the existing blocks of the output extent, allowing individual blocks on a track to be updated without erasing other blocks on that track.

Example 3. Physically copy tracks.

This example corrects the problem with the VTOC as described earlier.

```
                     // JOB P3
                     // EXEC DRD,SIZE=200K
set global options->    COPY=NOP,UPSI=111,PRINT=YES
copy one track----->    COPY=PHYSICAL,CVOL=CKD476,SBEGIN=4,RB=3,RM=S
                     /*
                     /&
```

***WARNING*: You should never copy the first track of the VTOC of one disk to the same track of a VTOC on another disk unless neither disk has alternate tracks assigned. If either disk has alternate tracks, it could result in two or more defective tracks assigned to the same alternate.**

Example 4. Physically copy tracks from a 3380 to a 3380.

This example copies tracks 100 through 119 from a 3380 on 131 to tracks 100 through 119 on a 3380 on 132.

```
                      // JOB P4
input disk--------->  // ASSGN SYS001,131
output disk-------->  // ASSGN SYS002,132
                      // EXEC DRD,SIZE=200K
set global options->     COPY=NOP,UPSI=011,PRINT=YES
copy the area------>     COPY=PHYSICAL,RMODE=IMAGE,CBEGIN=610,CHALT=714
                      /*
                      /&
```

Example 5. Physically copy tracks from a 3350 to a 3350.

This example copies tracks 100 through 119 from a 3350 on 131 to tracks 120 through 139 on a 3350 on 132.

```
                      // JOB P5
input disk--------->  // ASSGN SYS001,131
output disk-------->  // ASSGN SYS002,132
                      // EXEC DRD,SIZE=200K
set global options->     COPY=NOP,UPSI=011,PRINT=YES
copy the disk area->     COPY=PH,RMODE=I,SB=310,SH=329,RB=400,RH=419
                      /*
                      /&
```

Example 6. Physical Save and Restore.

Assume a physical control file which does not have a label in the VTOC. This example saves the area on the back of a 3375 disk on 231. Then it restores the area to the end of a 3380 on 1C1. The file is accessed in a sequential manner to fit the maximum number of records per track on the 3380.

```
                      // JOB P6 SAVE
                      // ASSGN SYS007,280
                      // TLBL TAPEOUT,'PHYSAVE'
                      // ASSGN SYS001,131
                      // EXEC DRD,SIZE=200K
set global options->     SAVE=NOP,UPSI=111,PRINT=YES
save the disk area->     SAVE=PH,SB=19600,SH=19800,EOF=YES,SVOLID=CKD231
rewind the tape---->     SAVE=NOP,TAPE=REWIND
                      /*
                      /&

                      // JOB P6 RESTORE
                      // ASSGN SYS008,280
                      // TLBL TAPEIN,'PHYSAVE'
                      // EXEC DRD,SIZE=200K
set global options->     REST=NOP,UPSI=111,PRINT=YES
restore area------->     RE=PH,RB=39900,RH=40100,RMODE=S,RVOLID=CKD1C1
unload the tape---->     RE=NOP,TAPE=RUN
                      /*
                      /&
```

Example 9. Copy a Physical control file.

This examples copies a 'Physical' control file from a 3380 on 131 to one on 132.

```
                      // JOB P9 COPY
input disk--------->  // ASSGN SYS001,131
output disk-------->  // ASSGN SYS002,132
                      // EXEC DRD,SIZE=200K
set global options->     COPY=NOP,UPSI=011,PRINT=YES
copy the disk area->     COPY=PHY,CBEGIN=105,CHALT=9907,RMODE=IMAGE
                      /*
                      /&
```

Saving, Restoring, Copying VM Mini-Disk

Example 10. Physically save selected VM Mini disks.

This example saves areas of a disk containing VM mini disks. The disk is a 3390 VM sysres with a number of CMS mini-disks after the sysres area.

```
                      // JOB P10 SAVE VM MINI DISKS
                      // TLBL TAPEOUT
                      // ASSGN SYS007,280
                      // EXEC DRD,SIZE=200K
set global options->     SAVE=NOP,UPSI=111,PRINT=YES,CP=YES
save the disk area->     SAVE=PHYSICAL,SVOL=VMPRES,SBEGIN=27100,SH=45714
rewind the tape---->     SAVE=NOPERATION,TAPE=REW
                      /*
                      /&
```

Example 11. Physically restore selected VM Mini disks.

This example restores the selected mini disks from the tape saved in example 10. The mini-disk at cylinder 271-273 is restored to its original location, while the one cylinder mini at cylinder 288 is moved to cylinder 101, keeping its original virtual disk count areas as backed up.

```
                      // JOB P11 SELECTIVE RESTORE OF VM MINI DISKS
                      // TLBL TAPEIN
                      // ASSGN SYS008,280
                      // ASSGN SYS002,352
                      // EXEC DRD,SIZE=200K
set global options->     RESTORE=NOP,UPSI=011,PRINT=YES
restore------------->    RE=PHYS,RBEGIN=27100,RHALT=45714,AREA=SELECT
select cyl 271-273->     27100-27314
cyl 288 to cyl 101->     28800-28814  TO=K10100
end selection------>     /@
                      /*
                      /&
```

Save/Restore/Copy Physical Addon/Update

The SAVE/RESTORE/COPY=PHYSICAL method can also allow individual disk blocks to be gathered/added or updated. This can be very helpful when a need to repair/manipulate data/files arises. Because of the physical (as opposed to logical) nature of this function, it may be wise to test the outcome by first copying the receiving disk area to a scratch area, running the copy to the scratch area and then examining the affected disk blocks.

Example 12. Physically copy in addon mode (RM=A).

This example copies a single block from cylinder 6 head 1 record 5, adding it to cylinder 9 head 2 at record 7. The rest of the receiving track is erased. Using this method, it is possible to gather blocks from various disk locations and put them together at a desired location. If SHALT and RHALT are used, multiple blocks can be copied by one command.

```
                     // JOB P12 COPY
                     // EXEC DRD,SIZE=200K
set global options->    COPY=NOP,UPSI=011,PRINT=YES
copy the one block->    CO=PH,CV=CKD476,SB=601R5,RB=902R7,RM=ADDON
                     /*
                     /&
```

Example 13. Physically copy in update mode (RM=U).

This example copies the data in a single block at cylinder 6 head 1 record 5 into the single block at cylinder 9 head 2 record 7. No other blocks on the receiving track are affected. If SHALT and RHALT are used, multiple blocks can be updated by one command. If the copied block is larger than the updated block, only the data that will fit is copied. If it is smaller, then the remainder of the receiving block is left as found.

```
                     // JOB P13 COPY
                     // EXEC DRD,SIZE=200K
set global options->    COPY=NOP,UPSI=111,PRINT=YES
update one block--->    CO=PH,CV=CKD476,SB=601R5,RB=902R7,RM=UPDATE
                     /*
                     /&
```

Summary of Control Card Format for Physical (CYL-HEAD) Operations

Save an Area of a Disk:

CKD→

```
SAVE=PHYSICAL(,EOF=YES)(,SSYSNO=SYSXXX)(,SBEGIN=CCCHH)
(,SHALT=CCCHH)(,STLOWER=HH)(,SUPPER=HH)(,VM=YES)
```

FBA→

```
SAVE=PHYSICAL(,SSYSNO=SYSXXX)(,SBEGIN=BBBBBB)(,SHALT=BBBBBB)
```

Restore an Area Of a Disk:

CKD→

```
RESTORE=PHYSICAL,RMODE=X(,RSYSNO=SYSXXX)(,RBEGIN=CCCHH)
(,RHALT=CCCHH)(,RTLOWER=HH)(,RUPPER=HH)(,VM=YES)(,AREA=X)
```

FBA→

```
RESTORE=PHYSICAL(,RSYSNO=SYSXXX)(,RBEGIN=BBBBBB)(,RHALT=BBBBBB)
(,AREA=XX)
```

Copy an Area of a Disk:

CKD→

```
COPY=PHYSICAL,RMODE=X(,EOF=YES)(,SSYSNO=SYSXXX)(,RSYSNO=SYSXXX)
(,SBEGIN=CCCHH,RBEGIN=CCCHH )(,CSYSNO=SYSXXX)(,CBEGIN=CCCHH )
,SHALT=CCCHH,RHALT=CCCHH(,STLOWER=HH)(,RTLOWER=HH)(,SUPPER=HH)
(,RUPPER=HH)(,CHALT=CCCHH )(,CTLOWER=HH )(,CUPPER=HH )
```

FBA→

```
SAVE=PHYSICAL,RESTORE=PHYSICAL(,SSYSNO=SYSXXX)(,RSYSNO=SYSXXX)
(,SBEGIN=BBBBBB)(,RBEGIN=BBBBBB)(,SHALT=BBBBBB)(,RHALT=BBBBBB)
(,CBEGIN=BBBBBB,CHALT=BBBBBB  )
```

Default:

```
SSYSNO=SYS001
RSYSNO=SYS002
```

Standalone Restore Procedures

Standalone Restore- native or VM

The standalone restore function allows volume type backups (SAVE=CURRENT/ALL/PHYSICAL) to be restored under VM or in Native Mode. Multiple disks may be restored from one or more backup tape sets, using one or two tape drives. The 'SAVE=IPL' function is not required on each tape set- it may be loaded from any tape, and another tape mounted once the 'DOCTOR 35' message appears. Multiple disks may be restored and other backup types may be present, intermixed with the volume backups.

The following procedure is used:

1. Load the save tape on any tape drive, use any disk drive. (If running under VM, do not put disk and tape on same virtual channel.)
2. The standalone supervisor must operate in 3270 console mode if in native mode, either 3215 or 3270 mode under VM, or 3215 if running under VM/CMS. Use the VM 'TERM CON' command to change console mode under non-CMS VM.
3. Set the IPL load address in the load screen (or load dial/ switches if appropriate). Skip this step if restoring under VM.
4. Press the enter key/load key four times. (For some tape drives, three hits on the enter key works better). Since the tape has standard labels this will space the tape over the VOL, HDR(s),and TAPE MARK preceding the Dr.D standalone supervisor. For 4300 type systems, an error message will appear until the supervisor loads. If the 'LABEL=NO' parameter is used, only one load is needed since the VOL and HDR records are not present. If under VM, enter 'I CUU' as many times as needed to space over the tape labels (until the DOCTOR 35 message appears).
5. After waiting for load to complete, hit the PA1/enter key if using a 3270 type console, or hit the request key. If running under VM, this step is not needed.
6. Message 35 appears: <-DOCTOR 35 ENTER RESTORE DISK TYPE

   Reply with:

2311	for 2311 disk
2314	for 2314,2319 disk
3330	for 3330 disk
334A	for 3340 disk mod 35
334B	for 3340 disk mod 70
3350	for 3350 disk
3375	for 3375 disk
3380	for 3380 disk
3390	for 3390 disk
9340	for 9340 disk
FBA	for any FBA disk type

7. MESSAGE 997 APPEARS: <-DOCTOR 997 ENTER ALTERNATE TAPE CUU .. If you do not wish to use an alternate tape drive and will restore from the

drive you ipl'd, hit enter to continue. If you want to use the ipl drive as the primary and another drive as the alternate, enter just the CUU of the alternate drive and restore will begin from the ipl'd drive. If you wish to begin restoring from a drive other than the ipl drive enter either 'CUU,CUU' for the primary and alternate drives, or enter 'CUU,' to indicate that restoring is to be done using the one drive. Note that this can allow the ipl'd drive to be released for other use once restoring begins.

8. Message 36 appears: <-DOCTOR 36 ENTER RESTORE DISK CUU Enter the CUU of the disk to be restored to and overwritten by the saved disk. If the VOLSER of the restored disk is to be changed from that saved, enter CUU,VOLSER.
9. Message 996 appears:<-DOCTOR 996 ENTER VOLSER OF DISK TO RESTORE If the disk to be restored is not the next thing on the tape you must enter the VOLSER of the saved disk as a search argument. (Enter it anyway if you're not sure.)
10. Message 16 appears: DOCTOR 16 RESTORING TO PACK VOL SER NO.... Hit enter to RESTORE, type 'C' to terminate.
11. Message 66 appears: DOCTOR 66 SAVE TAPE CREATED ON..... Hit enter to RESTORE, type 'C' to terminate.
12. The RESTORE continues until an EOJ MSG appears: <-DOCTOR 40 VOLUME RESTORE COMPLETED-TYPE END OR HIT ENTER.. If another disk is to be restored, hit enter, or type 'END' to terminate processing. If you wish to restore from another tape, mount it on the drive currently in use. You may also type 'U' (for unload) if you want DR.D to unload the tape just completed before restoring from another tape you mount on the same drive.

***Warning* Remember that only SAVE=ALL/CUR/PHYS backups may be restored standalone.**

Example 1. Save DOSRES for Standalone Restore

This job stream saves the DOSRES for use in a possible standalone restore situation.

```
                        // JOB EXAMPLE 1A STANDALONE WITH TAPE MANAGER
first file on tape--->  // TLBL TAPEOUT,'DRD.STANDALONE',,,,1
2nd file on tape----->  // TLBL BACKUP,'DOSRES.BACKUP',,,,2
                        // EXEC DRD,SIZE=200K
DRD programs to tape->     SAVE=IPL
set global options--->     SAVE=NOP,UPSI=111,PRINT=LIST,CP=YES
save the disk-------->     SAVE=CUR,SV=DOSRES,EX=YES,TD=BACKUP
                        /*
                        /&

                        // JOB EXAMPLE 1B STANDALONE WITHOUT TAPE MANAGER
                        // ASSGN SYS007,280
                        // MTC REW,SYS007
                        // TLBL TAPEOUT
                        // EXEC DRD,SIZE=200K
DRD programs to tape->     SAVE=IPL
set global options--->     SAVE=NOP,UPSI=111,PRINT=LIST,CP=YES
save the disk-------->     SAVE=CUR,SV=DOSRES,EX=YES
unload the tape------>     SA=NOP,TAPE=RUN
                        /*
                        /&
```

Note: No other control statements should precede the 'SA=IPL' and there should be no other parameters specified on the same line as the 'SA=IPL' control card.

See section "Restoring disk volumes from tape." on page 52 For NON-STANDALONE restore of standalone tape

Note: Whenever possible the restore should be run under VM or VSE to insure the greatest possible recovery from non-permanent disk errors.

Standalone restore under VM/CMS

The following job stream creates a "SAVE=IPL" tape containing the DRD standalone supervisor and restore programs. The tape is punched into TEXT cards into the VM reader of the user (JIM in the example). The reader file can then be "received" into CMS, then renamed to be DRDVMIPL TEXT A1. Once this is done, a CMS LOAD and START can be used to execute the DRD standalone restore. 3215 TERM CON and MACHINE (370) must be in effect.

```
                        * $$ JOB JNM=DRDVMIPL,DISP=D,CLASS=0
                        * $$ LST CLASS=X,DISP=K,JSEP=0
                        * $$ PUN CLASS=V,DEST=(*,JIM),JSEP=0
                        // JOB DRDVMIPL PUNCH SA=IP TAPE INTO TEXT CARDS
                        // TLBL TAPEOUT
                        // TLBL TAPEIN
assgn scratch tape---> // ASSGN SYS007,????
                        // MTC REW,SYS007
                        // EXEC DRD,SIZE=200K
put DRD on tape------>    SAVE=IPL
                        /*
                        // PAUSE PUNCH TEXT CARDS TO VM USER
                        // ASSGN SYS008,SYS007
                        // MTC REW,SYS008
punch TEXT cards-----> // EXEC DRDPUNVM,SIZE=600K
                        /*
                        /&
                        * $$ EOJ
```

Volume Logical

Disk Pool Backup and Restore

The Dr.D Volume Logical Function allows non-VSAM files to be saved or restored on a logical file basis without having to specify either the disk or disk track/block address, allowing users to selectively save files on one or more disks in a manner that will allow individual files to be selectively restored. For save logical VTOC, up to 30 volsers may be entered to designate the pool of disks from which individual files are to be saved. Users may also reorganize the files and/or the entire disk or pool of disks. The files saved by this method may also be restored as specified in section 4. The following section provides general information about the use of logical volume operations.

When Saving, the 'SV=' parameter must be entered and if restoring, both 'SV=' and 'RV=' must be specified to allow processing and distinguish VTOC from VSAM Catalog processing. During Restore, the 'SV' is required for tape positioning. In either case, the 'VO' (Volselect) may be used to select the volume(s) used to determine whether a file is to be Saved/Restored, as follows: If saving, any file that fully or partially resides on a selected volume is saved in its entirety. If restoring, the 'VO' refers to the Volume Serial(s) of the saved disk(s), and the selection logic is the same as when saving.

Starting with Dr.D version 6.5.1, files are saved in file-id sequence instead of the unpredictable order as found in the VTOC(s)

The files saved by this method may also be restored as specified in section 4. The following section provides general information about the use of the logical volume operations.

When Saving, the 'SV=' parameter must be entered and if restoring, both 'SV=' and 'RV=' must be specified to allow processing and distinguish VTOC from VSAM Catalog processing. During Restore, the 'SV=' is required for tape positioning. In either case, the 'VO=' (VOLSELECT) may be used to select the volume(s) used to determine whether a file is to be Saved/Restored, as follows: If saving, any file that fully or partially resides on a selected volume is saved in its entirety. If restoring, the 'VO=' refers to the Volume Serial(s) of the saved disk(s), and the selection logic is the same as when saving.

The first backup file contains file information extracted from the VTOC(s), and is used to generate label information for the restore.

During the restore operation, the beginning relative track number for the generated extent cards can be set through use of the 'EB=NNNNN' (Extent Begin) parameter. One purpose of this is to set the 'trigger' for disk managers since each generated extent begins at the value entered, thus allowing the disk manager to allocate space for files being restored.

Data secured files and VSAM data spaces are automatically excluded from backup unless included by 'FSECURED=YES' or by 'VSAM=YES' being specified.

(VSAM files should normally be saved as VSAM files using 'SAVE=LOGICAL' VSAM or 'SAVE=VSAM', to have any real usefulness.)

During the backup, the files being saved may be limited through use of the 'EXPIRED=' parameter, the 'PARTIAL=YES' parameter, the 'VOLSEL=' parameter, or the 'AREA=' parameters.

The 'EX=' parameter may be used in four different ways:

1.	EXPIRED=YES	Saves all files (the default value).
2.	EXPIRED=NO	Saves only those files that are unexpired when compared to the IPL date.
3.	EXPIRED=+/-NNNNN	Saves only those files that are unexpired when compared to the IPL DATE +/-NNNNN.
4.	EXPIRED=YYDDD	Saves only those files that are unexpired when compared to the Julian date (YYDDD).

The 'PARTIAL=SELECT' parameter allows files to be selected during either the backup or restore operation. The files may be selected on either an individual or generic (group) basis. The control cards for the selection follow the function control card, with the desired file I.D. in 1-44. You may also use the 'DD=' parameter to specify a file to be selected/rejected by the file id in a DLBL statement. The 'FILESIZE=' selector may also be used with 'PARTIAL=' to select files by their sizes (see 'FILESIZE=' in the keyword section).

To select files on a generic basis, replace any I.D. positions to be ignored with a null character. The assumed null character is the '/', but can be set to any value through use of the 'NULL=X' parameter. If an asterisk (*) is coded, the portion of the I.D. to the right of that position is considered null.

Files may also be selected by scanning the ID for a specified string by coding '=S=' in columns 1-3 preceding the search string field. Additionally, the effect of any selector card can be reversed by coding '@REVERSE@' to the right of the selection data (with at least one space separating), i.e., if the effect of the selector card is to select the file, it would instead be rejected and vice versa. Remember, however, that the first match of an ID to a selector card is used to either select or reject a file, so enter them in proper order to get the desired effect.

'PARTIAL=REJECT' allows the selective rejection of files on the same basis as 'PA=SE'.

The 'AR=' parameter allows you to limit the area of the disk(s) being saved or restored. If saving, only files whose first/only extent is within the specified disk limits are included in the file selection process. If restoring, this parameter indicates that a reorganizing restore is to be done, and specifies the disk area(s) to contain the restored files.

If restoring under the control of a disk manager, use the 'DM=' parameter to allow the disk manager to allocate file space, and the 'PI' (Pool ID) to specify the pseudo volser for disk manager use. Do not specify 'DM=' unless the disk manager is to allocate the space.

The 'DC=' (Dr.D Control) parameter is used to pass additional control parameters to Dr.D during the save/restore operation. The information within the parentheses is appended to the generated control card and is not checked until save/restore begins. By specifying an asterisk after the last comma delimiter, you can continue the control statement on another card, but the 'SV=' parameter must be entered on the first control card of the group and 'PARTIAL' parameters must be entered on the last control card of the group.

Example 1. Save all files.

This example saves all the files except those files that are expired, on the disks having volser 111111 and 222222.

```
// JOB VL1
// ASSGN SYS007,280
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
SAVE=LOGICAL,UPSI=111,SVOLSER=(111111,222222),EXPIRED=NO
SAVE=NOP,TAPE=REW
/*
/&
```

Example 2. Restore after interruption.

Restores the tape produced in Example 1. This example assumes the restore was interrupted after restoring 15 files. It now begins with the sixteenth restore.

```
// JOB VL2
// TLBL TAPEIN
// ASSGN SYS008,280
// ASSGN SYS002,131
// EXEC DRD,SIZE=200K
RESTORE=LOGICAL,SVOLSER=111111,RVOLSER=111111,SKIP=015
RESTORE=NOP,TAPE=REW
/*
/&
```

Example 3. Restore specific file.

Restores from the tape produced in Example 1. This example restores the file with I.D. 'DATA.FILE1' and any files starting with 'GL.SYS', but does not restore 'GL.SYS.GARBAGE'. It starts each extent at relative track 1 to allow a disk management system to allocate disk space.

```
// JOB VL3
// TLBL TAPEIN
// ASSGN SYS008,280
// EXEC DRD,SIZE=200K
RESTORE=LOGICAL,SVOL=111111,RVOL=111111,PARTIAL=SELECT,EBEGIN=1
DATA.FILE1
GL.SYS.GARBAGE      @REVERSE@
GL.SYS*
/*
RE=NO,TA=RE
/*
/&
```

Example 4. Restore of files from one volume to a contiguous area on another volume.

From the tape produced in the first example, this example restores the files saved from the disk on 132 (VOLSER 222222) to another disk on 133 (VOLSER 333333). It then restores the files to the contiguous area starting with track 12.

```
// JOB VL4
// TLBL TAPEIN
// ASSGN SYS008,280
// EXEC DRD,SIZE=200K
RESTORE=LOGICAL,SVOL=111111,VOLSEL=222222,RVOL=333333,EBEGIN=R12
RESTORE=NOP,TAPE=REW
/*
/&
```

Example 5. Save files less than a week old.

Save all the files from disk volser 111111 except those that will be expired seven days from the IPL date.

```
// JOB VL5
// ASSGN SYS007,280
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
SAVE=LOGICAL,SVOLSER=111111,UPSI=111,EXPIRED=+7
SAVE=NOP,TAPE=REW
/*
/&
```

Example 6. Single file restore.

Restore file 3 from the volume backup using the single file type Dr.D operation.

```
// JOB VL6
// TLBL TAPEIN
// ASSGN SYS002,131
// ASSGN SYS008,280
// DLBL FILE3,'FILE3'
// EXTENT SYS002,222222,1,0,20,1000
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILENAME=FILE3,SKIP=002
RESTORE=NOP,TAPE=REW
/*
/&
```

NOTE: The first backup file contains VTOC file information. All files following this file are user files.

Example 7. Save selected files.

Save the files residing in a pool of disk space on three 3380 disks having volume serial numbers 'VOLSR1,VOLSR2, and VOLSR3'. Save only the files that fully or partially reside on 'VOLSR2, and VOLSR3'.

```
// JOB VL7    SAVE SELECTIVELY FROM POOL OF DISK FILES
// ASSGN SYS007,181
// MTC REW,SYS007
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
SAVE=NOPERATION,UPSI=111
SAVE=LOGICAL,SVOL=(VOLSR1,VOLSR2,VOLSR3),VOLSEL=(VOLSR2,VOLSR3)
SAVE=NOPERATION,TAPE=REW
/*
/&
```

Example 8. Save selected files.

Save the files residing in a pool of disk space on three 3380 disks having volume serial numbers 'VOLSR1,VOLSR2, and VOLSR3'. The space on the first disk is in cylinder 1 through 100, the second and third, 100 through 600.

```
// JOB VL8    SAVE POOL OF DISK FILES
// ASSGN SYS007,181
// MTC REW,SYS007
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
SAVE=NOP,UPSI=111
SAVE=LOG,SV=(VOLSR1,VOLSR2,VOLSR3),AREA=(15-1500,1500-9000,R)
SAVE=NOP,TAPE=REW
/*
/&
```

Note: That the 'AR' ranges are stated in tracks or FBA blocks, and the cylinder addresses have been converted accordingly.

Example 9. Restore and Reorganize.

This example restores and reorganizes the pool of disk space saved in example eight, keeping each extent of each file on its original disk. It changes the expiration date of each file being replaced on disk so that no 'overlap' messages occur and no operator intervention is needed.

```
// JOB VL9   REORGANIZE DISK POOL
// ASSGN SYS008,181
// MTC REW,SYS008
// TLBL TAPEIN
// EXEC DRD,SIZE=200K
RESTORE=NOP,UPSI=111
RESTORE=LO,SV=VOLSR1,RV=(VOLSR1,VOLSR2,VOLSR3),DELETE=YES,*
AREA=(15-1500,1500-9000,R),EBEGINEXTENT=RS
RESTORE=NOP,TAPE=REW
/*
/&
```

Example 10. System Manager restore.

Restore the files from Example 8. Let System Manager allocate the files during the restore. The files were saved from a 3340 disk and are being restored to space on CKD 3380 disks, so the tracks to be allocated must be adjusted using the 'CF' parameter. We know that five 3340 track length records can fit on each 3380 track, so a ratio of one to five is needed to adjust the 3380 allocation, or 'CF=020'.

```
// JOB VL10  SYSTEM MANAGER RESTORE
// ASSGN SYS008,181
// MTC REW,SYS008
// TLBL TAPEIN
// DLBL SYSIN,'IJSYSIN',NEW,TRACKS=15,SYS001
// EXEC DRD,SIZE=200K
RESTORE=NOP,UPSI=111
RESTORE=LOGICAL,SVOL=VOLSR1,RVOL=VOLSR1,DMAN=SM,CFACTOR=020,*
PID=VOLSR1
/*
// ASSGN SYS000,SYSRDR
// MTC REW,SYS008
// TLBL TAPEIN
// DLBL IJSYSIN,'IJSYSIN',OLD,SYSIN
```

Note: that the 'PID=' (POOL ID) parameter is used to direct the files to a particular disk (V=VOLSR1 is added to the DLBL(s)).

File Rename

During restore, file IDs may be altered in conjunction with the PARTIAL=SE/RE' parameter. Several methods are available. The IDs may be fully replaced, transformed, prefixed, or any combination, by the addition of rename control cards following the selector card used to select the file for restore. The three possibilities are as follows:

1. Starting in column 1, code '=T=' followed by a search string, an '=', and a replace string (or space if the search string is to be removed). Only the first occurrence of the string is replaced, so use multiple '=T=' cards if you wish to transform multiple occurrences.
2. Starting in column 1, code '=P=' followed by a string to be added to the front of the generated ID.
3. Code the replacement ID starting in column 1 and '@RENAME@' to the right of the replacement (separated by at least one space). Use the '*' to halt character replacement, or the '/' to prevent replacement of individual characters.

Example 11. File Rename restore.

Restore the files from Example 8. The files are to be used for testing, so rename them appropriately, and restore them to a test disk, starting at track 15.

```
// JOB VL11            FILE RENAME RESTORE
// ASSGN SYS008,181
// MTC REW,SYS008
// TLBL TAPEIN
// EXEC DRD,SIZE=200K
RESTORE=NOP,UPSI=111
RESTORE=LOG,SVOL=VOLSR1,RVOL=VOLSRT,EBEGIN=R15,PARTIAL=SELECT
=S=PAYROLL
=P=TEST.
=S=COST
=T=COST=CA
=P=TEST1.
/*
RESTORE=NOP,TAPE=REW
/*
```

Example 12. File Rename restore.

Restore the files from Example 8. The files are to be used for testing, so rename them appropriately, and restore them to a test disk, starting at track 15.

```
// JOB VL12            FILE RENAME RESTORE
// ASSGN SYS008,181
// MTC REW,SYS008
// TLBL TAPEIN
// EXEC DRD,SIZE=200K
RESTORE=NOP,UPSI=111
RESTORE=LOG,SVOL=VOLSR1,RVOL=VOLSRT,EBEGIN=R15,PARTIAL=SELECT
PAYROLL*
=P=TEST.
COSTS*
BUCKS*          @RENAME@
=P=TEST1.
/*
/*
/&
```

Summary of Control Card Formats for Volume Logical Operations

Logical Save/Restore from a Volume.

```
SAVE/RESTORE=LOGICAL
        AREA=(S1-E1,S2-E2,R),
        DELETE=YES,
        DMANAGER=X,
        EBEGIN=NNNNNN, OR EBEGIN=RNNNNNN, OR EBEGIN=RS
        CF=NNN,
        FSECURED=YES,   (NO ASSUMED)
        NULL=X,
```

```
PARTIAL=SELECT/REJECT
RVOLSER=SSSSSS, OR RVOLSER=(VOLSN1,VOLSN2,...),
SVOLSER-SSSSSS, OR SVOLSER=(VOLSN1,VOLSN2,...),
VOLSEL=SSSSSS, OR VOLSEL=(VOLSN1,VOLSN2,...),
VSAM=YES, (NO ASSUMED)
```

Disk Reorganization: The RESTORE/LOGICAL (VTOC) feature allows the elimination of disk fragmentation and/or relocation of files by three methods:

1. Full Reorganization: Each file is restored contiguously starting with the first (or only) disk, filling each disk (or disk area) as fully as possible. This method is invoked by specifying either 'EB=Rnnnnnn' for single volume restore or 'AR=' for multiple volume restores. (This method may leave existing file extents in unused areas of a disk pool.)
2. Full In Place Reorganization: All disks in the pool are reorganized while keeping each file's extents in place on its original disk(s). This method is invoked by specifying 'EB=RS' and 'AREA='.
3. Partial In Place Reorganization: Selected disks are reorganized keeping each file's extents in place on its original disk(s). This method is invoked by specifying 'EB=RS', 'AREA=', and 'VOLSEL='.

NON-VSAM SD and DA Data Files

Saving, Restoring, Copying SD, DA Data files

To work with a data file, specify 'SAVE/COPY/RESTORE=DATA'. 'SFILENAME=' must always be specified when 'SAVE/COPY=DATA' to inform Dr.D of the file to be read. Likewise, 'RFILENAME' must always be specified when 'COPY/RESTORE=DATA'. Additionally DLBL and extents must be provided for either in the standard labels or JCL for those files. If 'COPY=DATA', both 'SF=' and 'RF=' must be specified and they must be unique.

Dr.D will interrogate the DLBL to ascertain whether it is SD, DA, or ISAM. For CKD disk, an 'SD' file may be restored as a 'DA' and vice versa. The restore file's extents are completely independent of the save file's- they may differ in size, placement, split file's, split cylinder specifications, and types of disks.

Usually all sequential files and some direct access files have an EOF record after the last data record of the last prime data extent. If this is so, specifying 'EOF=YES' will cause Dr.D to stop saving after reading an EOF record. This will result in a faster save since that entire extent does not need to be saved.

For 'SD' files, a logical record count may be obtained through use of the 'LR=' (LOGICAL RECORD LENGTH) and the '// UPSI XX1'. For variable blocked files, only the upsi is required.

When restoring 'SD' files to FBA disk, specify 'LR=XXXX' (logical record length) for fixed length records. If this is not done, fixed length records will be restored as undefined. For variable blocked records, use 'LB=' (LOGICAL BLOCK=nnnnn max variable block).

If restoring to CKD disk, Dr.D uses physical IOCS (PIOCS) to write the data to disk. This may not be satisfactory if a disk manager is installed for several reasons: file truncation may not occur, and the disk manager catalog may not be updated. To overcome this problem, use 'RESTORE=SD' instead of 'RESTORE=DA'. This will cause a DTFSD to be used during the restore/copy, allowing the disk manager to function normally when the file is closed. This will also allow file reblocking if the 'LB' or 'LR'/'NR' parameters are used.

Example 1. Save 'SD' and 'DA' files.

This example saves an 'SD' and a 'DA' file to tape. It also generates a logical record count for the 'SD' file.

```
// JOB D1
// DLBL SEQFILE,'DISK FILE 01',99/360,SD
// EXTENT ,737723,1,0,20,3980
// EXTENT ,737724,1,1,20,2000,8
// DLBL RANFILE,'DISK FILE 02',99/360,DA
// EXTENT ,434729,1,0,20,2320
// TLBL TAPEOUT,'SAVE TAPE'
// ASSGN SYS007,280
// EXEC DRD,SIZE=200K
SAVE=NOP,PRINT=YES,UPSI=111
SAVE=DATA,SFILE=SEQFILE,EOF=YES,LRECORD=100,CPRESS=YES
SAVE=DATA,SFILE=RANFILE
/*
/&
```

Example 2. Restore 'SD' and 'DA' files.

This example restores the files from Example 1.

```
// JOB D2
// DLBL FILEONE,'DISK FILE 01',99/360,SD
// EXTENT ,737723,1,0,20,3980
// EXTENT ,737724,1,1,20,2000,8
// DLBL FILETWO,'DISK FILE 02',99/360,DA
// EXTENT ,434729,1,0,20,2320
// TLBL TAPEIN,'SAVE TAPE'
// ASSGN SYS008,280
// EXEC DRD,SIZE=200K
RESTORE=NOP,PRINT=YES,UPSI=111
RESTORE=DATA,RFILE=FILEONE
RESTORE=DATA,RFILE=FILETWO
/*
/&
```

Summary of Control Card Formats for Data Files (SD,DA)

Save Data File (SD or DA)

```
SAVE=DATA,SFILE=FFFFFFF(,EOF=YES)(,CPRESS=YES)
```

Restore Data File (SD or DA)

```
RESTORE=DATA,RFILE=FFFFFFF(,R0=YES)
```

Copy Data File (SD or DA)

```
SAVE=DATA,RESTORE=DATA,SFILE=FFFFFFF,RFILE=FFFFFFF(,EOF=YES)
        (,R0=YES) (  COPY=DATA  )
```

Added Facility for SD and DA FILES

Using The 'RMODE' Operand

The 'RMODE' (Restore Mode) is an internal switch in Dr.D that is set and used when writing to the disk. Set 'RMODE' to either an 'I' for image or 'S' for squeeze.

When set to 'I', Dr.D exactly mirrors the saved extent during the restore. Partially used tracks, unused tracks, records out of sequence on a track, all appear restored in the same format.

When restoring a DA data file, 'RMODE' is internally set to 'I' to accommodate just these types of things which are likely to occur in a DA file. Thus, if relative track addressing is used, the DA file may be relocated and still accessed. The DA file may also be exactly mirrored on another type of disk, and changing the 'DTF', 'SELECT', or 'WHATEVER' in a program - may still be accessed.

When RMODE is set to 'S', the full track capacity of the restore disk is employed (Squeeze). If there are any partial or unused tracks in the saved extent, that unused area is dropped and ignored. As many data records as can be accommodated are written on each track. Also, any records that are out of sequence on a saved track are written in the physical order they appear on the saved track. However, all records written on a track are sequentially numbered.

When restoring an SD file, RMODE is set to 'S' if it is restored to a new disk type. This operation allows the file to be moved to another type of disk and still be accessed by the same program after changing the 'DTF', 'SELECT', or 'WHATEVER' for the new device. The reason behind this digression is that exceptional conditions may arise where you might need to change the assumed default. Knowing how and what each does aids in deciding on which code to use.

Specifying 'RM=I or S' on the CTL card will override the assumed default. The only reason for specifying this override is in conversion of direct access from one disk type to another.

Example 1. Transfer DA file.

Example 1 assumes a DA file organized in the following manner: While the records are randomly scattered throughout the disk, the entire extent has been, in fact, originally pre-formatted with 'X' number of 'EMPTY' blocks per track. As a result, there are only 'EMPTY' blocks containing no data. Each record is accessed by randomizing its key to a relative track and record address. One file resides on a 3350. The example transfers it to a 3380.

Note: If 'RM=S' is not on the control card, it will produce the same number of blocks per track on both devices. This program is modified to use the full capacity of the 3380 track. As a result, the example squeezes as many blocks per track as possible onto the 3380.

```
// JOB R1 TRANSFER DA FILE
// ASSGN SYS001,132
// ASSGN SYS002,1C2
// DLBL F3350,'F3350',99/365,DA
// EXTENT SYS001,111111,1,0,20,380
// DLBL F3380,'F3380',99/365,DA
// EXTENT SYS002,111111,1,0,19,380
// EXEC DRD,SIZE=200K
COPY=DATA,SFILE=F3350,RFILE=F3380,RMODE=SQUEEZE
/*
/&
```

Further Note: When converting SD files from one disk type to another the 'EOF' parameter must be used to insure that the end of file record is positioned properly. If 'EOF=YES' is used, Dr.D will write an 'EOF' after the last block and on the next track (unless the last block is on the last track). Some languages/access methods will detect the EOF only if it is written in a location which would accommodate a block, even though an EOF would fit.

Summary of Control Card Format for RMODE Operand

Alter RMODE

```
RESTORE=DATA(,...ETC...),RMODE=SQUEEZE
COPY=DATA           RMODE=IMAGE
```

Added Facility for SD and DA Files Using the 'SYNC=YES' Operand.

Dr.D processes multi-extent data files as normally expected under DOS. The size and number of extents of the save file and restore file are independent of each other. Upon restoring, Dr.D will fill each extent in order, switching to the next extent as the current one becomes full (the input and output extents are treated asynchronously).

Example 1. Copy Three extents to One.

This example creates a job stream to copy a three extent file over to a one extent file.

```
// JOB S1 MERGE THREE EXTENTS TO ONE
// DLBL THREE
// EXTENT SYS001,111111,1,0,20,20
// EXTENT SYS001,111111,1,1,40,20
// EXTENT SYS001,111111,1,2,60,20
// DLBL ONE
// EXTENT SYS001,111111,1,0,1000,60
// EXEC DRD,SIZE=200K
COPY=DATA,SFILE=THREE,RFILE=ONE
/*
/&
```

Example 2. Split One extent into Three.

This example splits a one extent file into three.

```
// JOB S1 SPLIT ONE EXTENT INTO THREE
// DLBL ONE
// EXTENT SYS001,111111,1,0,1000,60
// DLBL THREE
// EXTENT SYS001,111111,1,0,20,20
// EXTENT SYS001,111111,1,1,40,20
// EXTENT SYS001,111111,1,2,60,20
// EXEC DRD,SIZE=200K
COPY=DATA,SFILE=ONE,RFILE=THREE
/*
/&
```

However, sometimes you might want to switch output extents before each fills up. Putting 'SYNC=YES' on a Restore/Copy for an 'SD or DA' data file will cause the save and restore files' extents to become 'SYNCHRONIZED'.

If there were 100 and 200 records respectively in the first and second extents of a save file, then there will be exactly 100 and 200 records, respectively, in the first and second extents of the restore file (even though they might be able to hold more). Whenever 'SYNC=YES' is specified, the restore file must have as many extents as the save file and each restore extent must be able to accommodate at least as many records as its corresponding save extent. Putting 'SYNC=YES' for an ISAM file will be ignored.

Example 3. Transfer file to 3-extent.

Example 3 takes a 3-extent random file on a 3350 and transfers it over to three larger extents on a 3380. At the same time, it keeps all the records in their corresponding extents.

```
// JOB S2 COPY ILLUSTRATING USE OF SYNC=YES
// DLBL INPUT,'F3350',99/365,DA
// EXTENT SYS001,654321,1,0,20,100
// EXTENT SYS001,654321,1,1,120,100
// EXTENT SYS001,654321,1,2,220,100
// DLBL OUTPUT,'F3380',99/365,DA
// EXTENT SYS001,987654,1,0,19,114
// EXTENT SYS001,987654,1,1,133,114
// EXTENT SYS001,987654,1,2,247,114
// EXEC DRD,SIZE=200K
COPY=DATA,SFILE=INPUT,RFILE=OUTPUT,SYNC=YES
/*
/&
```

Summary of Control Card Format for SYNC Operand

Synchronize Output Extents with Input

```
RESTORE=DATA(,...ETC...),SYNC=YES
COPY=DATA
```

Dr.D LISTVTOC Function

The DR.D LISTVTOC function provides a compact (but complete) report of the files in the VTOC of one or more disks. This function is almost identical to the DR.D VTOC 'SAVE LOGICAL' feature with the same file selection capability. The only difference is that no files are saved, and 'FUNCTION=LVTOC' is specified instead of 'SAVE=LOGICAL'. Like 'SAVE LOGICAL', either single or multiple disks may be processed. For DR.D LISTVTOC, up to 30 volsers may be entered to designate the pool of disks from which individual files are to be listed. The report is produced in FILE ID sequence, with all files selected sorted together, allowing a complete alphabetic list of all files in a single / pool of disks.

At the end of file listing, a space map is printed showing the location and size of each available space on each disk in the pool of disks specified.

The FORMAT= keyword can be used to affect the report in several ways. If FORMAT=A is specified, no space map is printed. If FORMAT=B is specified, individual files are not listed. If an extent listing is desired, a two character operand is used where the second is an 'E', i.e. FORMAT=AE, FORMAT=BE OR FORMAT=XE (if neither option 'A' OR 'B' is desired).

Example 1. List all VTOC entries of a disk.

```
// JOB DR.D LISTVTOC ALL
// EXEC DRD,SIZE=200K
FUNCTION=LVTOC,SVOLSER=DISK01
/*
/&
```

Example 2. List selected files.

```
// JOB DR.D LISTVTOC SELECTIVE
// EXEC DRD,SIZE=200K
FUNCTION=LVTOC,SVOLSER=DISK01,PARTIAL=SELECT
PAYROLL.SYS*
/*     TERMINATE SELECTION
/*     TERMINATE EXECUTION
/&
```

Example 3. List files associated with a disk.

```
// JOB DR.D LISTVTOC FILES USING DISK VOLSER CKD201
// EXEC DRD,SIZE=200K
FUNCTION=LVTOC,SVOLSER=(DISK01,CKD201),VOLSELECT=CKD201
/*
/&
```

Example 4. List no files, just the Space Map.

```
// JOB DR.D LISTVTOC NO FILES, JUST THE SPACE MAP
// EXEC DRD,SIZE=200K
FUNCTION=LVTOC,SVOLSER=DISK01,PARTIAL=REJECT
*       REJECT ALL FILES
/*      TERMINATE SELECTION
/*      TERMINATE EXECUTION
/&
```

VTOC Move Function

The usual method of VTOC relocation or resizing is to save the individual files, reinitialize the disk, and then restore the files. This function allows the VTOC of a disk to be quickly and easily moved to a new location and/or resized. All that is required is that any files in the specified location be deleted or moved to a new location, before the VTOC move is attempted. If the size of the VTOC is being reduced, check to see that no labels exist in the part of the VTOC being dropped. Before moving the VTOC, Dr.D checks the new location against the VTOC to be sure that no files (expired or unexpired) exist in the new location, and that the new location is within the disk limits and will not overlap either the VOL1 record or the old VTOC. If the size of the VTOC is being reduced, the part of the VTOC being dropped is checked to insure no loss of labels. If any conflict is detected, the job is canceled, otherwise the operator is informed that the VTOC is about to be moved or resized, and given one last chance to abort the procedure.

Example 1. Move VTOC on a CKD device from one area of a disk to another.

```
// JOB VTOC MOVE TO CYLINDER 500 HEAD 00
// EXEC DRD,SIZE=200K
   FUNCTION=VTOCMOVE,RVOLSER=CKD111,RBEGIN=50000
/*
/&
```

Example 2. Move VTOC on a FBA device to a new location.

```
// JOB VTOC MOVE TO FBA BLOCK 250,000
// EXEC DRD,SIZE=200K
   FUNCTION=VTOCMOVE,RVOLSER=FBA111,RBEGIN=250000
/*
/&
```

Example 3. Move and resize a VTOC on a CKD device.

```
// JOB VTOC MOVE TO CYLINDER 50, ADDING A CYLINDER
// EXEC DRD,SIZE=200K
   FUNCTION=VTOCMOVE,RVOLSER=CKD111,RBEGIN=5000,RHALT=5114
/*
/&
```

Example 4. Resize a VTOC on a CKD device.

```
// JOB REDUCE VTOC FROM 15 TRACKS TO 5 TRACKS
// EXEC DRD,SIZE=200K
   FUNCTION=VTOCMOVE,RVOLSER=CKD111,RBEGIN=88400,RHALT=88404
/*
/&
```

VTOC File Delete Function

The VTOC delete function allows files to be deleted using the same flexible selection process used in the 'SAVE/RESTORE=LOGICAL' and 'LISTVTOC' functions. This can be useful when reorganizing DISK space or when moving files from one DISK to another, or anytime single/groups of files are to be deleted.

To prevent the accidental deletion of files, this function uses a slightly different approach to file selection- no file is deleted unless 'PARTIAL=SELECT/REJECT' is entered. IF the 'PARTIAL' parameter is not entered, the only action taken is the listing of the files that would be deleted. Additionally, the 'FD=TEST' parameter should be used initially to produce a list of files to be deleted. Once the list has been checked for correctness, then 'FD=DELETE' can be used to actually delete the file(s).

Example 1. Test run of the DELETE function, (no files will be deleted).

The following example produces a list of files in a 'PAYROLL' system that have been moved to a new disk pool (using 'SAVE=LOGICAL' and 'RESTORE=LOGICAL'), and now must be deleted from the source disk(s).

```
// JOB DELETE (TEST RUN, NO FILES DELETED)
// EXEC DRD,SIZE=200K
   FUNCTION=DELETE,RVOLSER=DISK01,FDELETE=TEST,PARTIAL=SELECT
   PAYROLL*
   /*  TERMINATES FILE SELECTION
/*  TERMINATES EXECUTION
/&
```

After checking the file list, the job is resubmitted as follows:

```
// JOB DELETE FILES
// EXEC DRD,SIZE=200K
   FUNCTION=DELETE,RVOLSER=DISK01,FDELETE=DELETE,PARTIAL=SELECT
   PAYROLL*
   /*  TERMINATES FILE SELECTION
/*  TERMINATES EXECUTION
/&
```

Please note that any file can be deleted with this function, even if it is secured or is VSAM space (use VSAM=YES, FSECURED=YES), also that some VSE system files have VSAM VTOC designations.

Example 2. Test run of the DELETE function with REVERSE specified, (no files will be deleted).

The following example produces a list of files that have been slotted for deletion, the first run will not delete the files but will show the file id's that would have been if FD=D had been specified instead of FD=T.

```
// JOB TEST OF DELETE FUNCTION WITH REVERSE
// EXEC DRD,SIZE=200K
   FUNCTION=DELETE,RVOL=111111,PARTIAL=SELECT,FDELETE=TEST
   DATA.FILE1
   GL.SYS.PERMANENT    @REVERSE@
   GL.SYS*
   /@
/*
/&
```

After checking the file list, the job is resubmitted as follows:

```
// JOB TEST OF DELETE FUNCTION WITH REVERSE
// EXEC DRD,SIZE=200K
   FUNCTION=DELETE,RVOL=111111,PARTIAL=SELECT,FDELETE=DELETE
   DATA.FILE1
   GL.SYS.PERMANENT    @REVERSE@
   GL.SYS*
   /@
/*
/&
```

Please note that when using the REVERSE feature in conjunction with the delete, that all of the files that have GL.SYS in their file id's will be deleted, except for the one that has @REVERSE@ beside it. Also please note that reversal cards must precede other selectors.

Dr.D Punch Function

The Dr.D Punch function provides a quick and easy method for creating IDCAMS control statements for defining VSAM files, and DLBL and EXTENT JCL statements. In either case, the statements are 'punched' to the punch queue where they may then be extracted for editing and further use. This may be particularly useful if the statements are not available for some reason and need to be recreated, or in the case of VSAM where a file needs to be defined that is similar to an existing file.

This function is almost identical to the Dr.D 'SAVE=LOGICAL' feature with the same file selection capability. The only difference is that no files are saved, and 'FUNCTION=PUNCH' is used instead of 'SAVE= LOGICAL'.

Example 1. Punch IDCAMS definitions selectively.

```
// JOB DR.D PUNCH SELECTIVE
// EXEC DRD,SIZE=200K
FUNCTION=PUNCH,SCATDDNAME=IJSYSCT,PARTIAL=SELECT,ASSOC=YES
COBOL.TEST*
PAYROLL.SYS*
/*     TERMINATE SELECTION
/*     TERMINATE EXECUTION
/&
```

Example 2. Punch IDCAMS definitions for a full catalog.

```
// JOB DR.D PUNCH ENTIRE CATALOG
// EXEC DRD,SIZE=200K
FUNCTION=PUNCH,SCATDDNAME=IJSYSCT
/*
/&
```

You may also Punch DLBL and EXTENT JCL statements from the VTOC of one or more disks. This is almost identical to 'SAVE=LOGICAL' VTOC with the same file selection capability. The only difference is that no files are saved and 'FUNCTION=PUNCH' is used instead of 'SAVE=LOGICAL'.

Example 3. Punch the VTOC of a selected disk.

```
// JOB PUNCH THE VTOC OF A SELECTED DISK
// EXEC DRD,SIZE=200K
FUNCTION=PUNCH,SVOLSER=111111
/*
/&
```

Example 4. Punch only files associated with a disk.

```
// JOB PUNCH ONLY FILES USING DISK VOLSER=CDK201
// EXEC DRD,SIZE=200K
FUNCTION=PUNCH,SVOLSER=(DISK01,CKD201),VOLSELECT=CDK201
/*
/&
```

Example 5. Punch selected files.

```
// JOB DR.D PUNCH SELECTED FILES
// EXEC DRD,SIZE=200K
FUNCTION=PUNCH,SVOLSER=DISK01,PARTIAL=SELECT
PAYROLL.SYS*
/*     TERMINATE SELECTION
/*     TERMINATE EXECUTION
/&
```

MINI-DISK Create Function

The usual method of MINI-DISK creation requires that the DSF utility be used to initialized the mini, a function not currently in VSE. This function eliminates that limitation by creating the mini-disk directly under VSE.

The real disk that is to contain the new mini-disk(s) must be attached to VSE in read/write mode and be in a device-up state so that an assignment can be made to it by VSE. The user must define the mini-disk 'real' start and end addresses via the RBEGIN and RHALT keywords. A VSE/MVS type VTOC is created at the end of the disk with a one cylinder size for CKD or a one CI size for FBA. The VOLSER of the created mini-disk is specified via the 'VOLOUT' keyword. If the size or location of the created mini-disk VTOC is inadequate, use the FUNCTION=VTOCMOVE to adjust it once the mini is attached to VSE.

For this function to be most effective, the VSE system destined to use the created mini-disk should have a number of 'dummy' PUBS (physical unit blocks) that can be used for attaching the newly created mini-disk(s). The VM 'USERDIRECT' can be updated without a re-ipl of VSE, but if no spare PUBS are available, a re-ipl after update of the VSE 'ADD' statements would be required. If spare PUBS are available, the new mini-disk(s) can be linked to VSE, readied by use of the VSE 'DVCUP' command, and then be used by VSE (adjust the VTOC at that time if desired).

Before each mini-disk is created, the DOCTOR 190 warning message is issued, and an operator response is required before the mini is created. Double check the RV, RB, and RH parameters to avoid the accidental destruction of mini-disks or user files.

Example 1. Create four 100 cylinder mini-disk on a CKD device.

```
// JOB CREATE FOUR CKD MINI-DISKS
// EXEC DRD,SIZE=200K
   FUNC=MC,RVOLSER=CKD111,VO=MINI01,RBEGIN=00100,RHALT=10014
   FUNC=MC,RVOLSER=CKD111,VO=MINI02,RBEGIN=10100,RHALT=20014
   FUNC=MC,RVOLSER=CKD111,VO=MINI03,RBEGIN=20100,RHALT=30014
   FUNC=MC,RVOLSER=CKD111,VO=MINI04,RBEGIN=30100,RHALT=40014
/*
/&
```

Example 2. Create four 10,000 block mini-disk on an FBA DEVICE

```
// JOB CREATE FOUR FBA MINI-DISKS
// EXEC DRD,SIZE=200K
   FUNC=MC,RVOLSER=FBA111,VO=MINI01,RBEGIN=00020,RHALT=10019
   FUNC=MC,RVOLSER=FBA111,VO=MINI02,RBEGIN=10020,RHALT=20019
   FUNC=MC,RVOLSER=FBA111,VO=MINI03,RBEGIN=20020,RHALT=30019
   FUNC=MC,RVOLSER=FBA111,VO=MINI04,RBEGIN=30020,RHALT=40019
/*
/&
```

Example 3. Move and resize a mini-disk VTOC.

```
// JOB VTOC MOVE TO CYLINDER ZERO
// EXEC DRD,SIZE=200K
   FUNCTION=VTOCMOVE,RVOLSER=MINI02,RBEGIN=1,RHALT=6
/*
/&
```

VTOC Expiration Date Modify

This function allows the expiration date of selected files to be set to 99/366 so that the files will be considered permanent by the operating system. Files are selected using the same logic as the other VTOC driven functions, with one additional selection parameter- the 'DATE=' keyword can be used to specify a date for selecting files by their existing expiration date (99/365 is assumed). If a file's expiration date is equal to or greater than the 'DATE=', and any other selection criteria is satisfied, then the file is selected, and its expiration date is changed to 99366.

This function was created to solve the problem of files having expiration dates of 99/365 to indicate permanent, non-expiring. Those files will be considered expired in the year 2000 unless their dates are changed to 99/366.

Example 1. Update the expiration date of all files in a pool of disks.

The following example updates the expiration date of all files in the VTOCs of the disks in a disk manager pool defined by the POOL-ID POOL01. Any file having an expiration date greater than 99/364 has its expiration set to 99/366 (permanent non- expiring).

```
// JOB DATEFIX1
// EXEC DRD,SIZE=200K
   FUNCTION=DFIX,RV=POOL01,FSECURED=YES,VSAM=YES
/* TERMINATES EXECUTION
/&
```

Example 2. Update the expiration date of files expiring in 1999.

The following example updates the expiration date of all files in the VTOC of the disk SYSWK1. Any file having an expiration date equal to or greater than 99/001 has its expiration date set to 99/366 (permanent non-expiring).

```
// JOB DATEFIX2
// EXEC DRD,SIZE=200K
   FUNCTION=DFIX,RV=SYSWK1,FSECURED=YES,VSAM=YES,DATE=99001
/* TERMINATES EXECUTION
/&
```

Example 3. Update the expiration date of selected files.

The following example updates the expiration date of all files in the VTOCs of the disks DISK01 and DISK02, having file ids starting with 'PAYROLL' and expiration dates starting in 1990.

```
// JOB DATEFIX3
// EXEC DRD,SIZE=200K
   FUNCTION=DFIX,RV=(DISK01,DISK02),DATE=90001,PARTIAL=SELECT
   PAYROLL*
   /@
/* TERMINATES EXECUTION
/&
```

User Exits

User Exit Functions

Two types of user exits are provided by Dr.D- the first allows user programs to Inspect, Modify, or Delete logical records during backup or restore of files in logical record format. This method should be used in most situations. If records are to be created, see the following section.

User Exit Type 1.

Dr.D loads the user program into the partition (use an EXEC size large enough for both Dr.D and the user program) and before each logical record is processed by Dr.D, it is passed to the user exit along with three one-byte flags to indicate end of file, whether backup or restore is in operation, and a user supplied record delete flag. The EOF flag is set to 'E' to indicate end of file, while the function flag is set to 'B' to indicate backup or 'R' to indicate restore. On return to Dr.D, the user exit may place a 'D' in the delete flag to request that the current record be deleted. (If saving, the record is not written to the tape. If restoring, the record is not restored.)

Dr.D passes five address constants to the user exit as pointers to the logical record, the 'EOF' flag, the function indicator, the delete flag, and the logical record length. This means that the user exit must have a 'USING' or 'ENTRY' type statement that establishes the addressability of these five items.

If the user exit is written in ANS COBOL, the GSIDMNS0 module supplied on the DRD install tape must at link edit time be substituted for the IBM supplied ILBDMNS0 module. This is most easily achieved by adding an INCLUDE to the link edit. DO NOT REPLACE THE IBM MODULE!

Examples of User Exit Operations

Example 1. ANS COBOL

```
// JOB MYJOB
// OPTION CATAL
   PHASE USERPROG,*
// EXEC FCOBOL
       IDENTIFICATION DIVISION.
       PROGRAM-ID     USERPROG.
       AUTHOR.
       DATE WRITTEN.
       REMARKS.
       ENVIRONMENT DIVISION.
       INPUT-OUTPUT SECTION.
       FILE-CONTROL.
       DATA DIVISION.
       LINKAGE SECTION.
       01 LOG-REC.
          10  USER-DATA.
              15  USER-FILED1     PIC  ......
       01  EOF-FLAG               PIC  X.
       01  FUNC-IND               PIC  X.
       01  DELETE-FLAG            PIC  X.
       01  REC-LENGTH             PIC  S9(8) COMP.

       PROCEDURE DIVISION USING LOG-REC EOF-FLAG FUNC-IND
       DELETE-FLAG REC-LENGTH.
           IF EOF-FLAG = 'E' THEN GOTO END-OF-JOB.
           IF .........  THEN GO TO DELETE-RTN.

       DELETE-RTN.
           MOVE 'D' TO DELETE-FLAG.  GOBACK.
       END-OF-JOB.  GOBACK.
/*
   INCLUDE GSIDMNS0
// EXEC LNKEDT
```

Example 2. Run JCL Example.

This example saves and flags a record for deletion with the use of user exit processing.

```
// JOB U16 BACKUP WITH DELETE
// DLBL A.'VSAM.FILE'..VSAM,CAT=IJSYSCT
// TLBL TAPEOUT
// ASSGN SYS007,...
// EXEC DRD,SIZE=200K
SAVE=NOP,UPSI=111,BLOCKSIZE=65496
SAVE=VSAM,SFILE=A,VARIABLE=YES,FAST=YES,USEREXIT=NF/USERPROG,*
DTAPE=YES
SAVE=NOP,TAPE=REWIND
/*
/&
```

Example 3. Job save with User Exit.

This example saves three VSAM files, with user exit processing on the 2nd and 3rd file. The user exits are named 'USER1' and 'USER2'.

```
// JOB SAVE WITH USER EXIT
// TLBL TAPEOUT
// ASSGN SYS007,...   ASSGN SYS007 TO TAPE IF NECESSARY
// DLBL A,'ORDER.DATA.BASE',,VSAM
// DLBL B,'MASTER.CARTON.DATA.BASE',,VSAM
// DLBL C,'FILLED.ORDER.DATA',,VSAM
// EXEC DRD,SIZE=300K
SAVE=NOP,UPSI=111,PRINT=YES,BLOCKSIZE=65496
SAVE=VSAM,SFILE=A,VARIABLE=YES,FAST=YES,AUTO=YES
SAVE=VSAM,SFILE=B,VA=YES,FAST=YES,AUTO=YES,USEREXIT=NF/USER1
SAVE=VSAM,SFILE=C,VA=YES,FAST=YES,AUTO=YES,USEREXIT=NF/USER2
SAVE=NOP,TAPE=REWIND
/*
/&
```

Example 4. Job restore with User Exit.

Example 2 restores the files saved in Example 1 with identical user exit processing.

```
// JOB RESTORE WITH USER EXIT
// TLBL TAPEIN
// ASSGN SYS008,...   ASSGN SYS008 TO TAPE IF NECESSARY
// DLBL A,'ORDER.DATA.BASE',,VSAM
// DLBL B,'MASTER.CARTON.DATA.BASE',,VSAM
// DLBL C,'FILLED.ORDER.DATA',,VSAM
// EXEC DRD,SIZE=300K
RESTORE=NOP,UPSI=111,PRINT=YES,BLOCKSIZE=65496
RESTORE=VSAM,RFILE=A,AUTO=YES
RESTORE=VSAM,RFILE=B,AUTO=YES,USEREXIT=NF/USER1
RESTORE=VSAM,RFILE=C,AUTO=YES,USEREXIT=NF/USER2
RESTORE=NOP,TAPE=RUN
/*
/&
```

Expanded function user exit

If the user exit is to generate some or all of the records to be restored by Dr.D, the following user exit format should be used. Two basic types of user exits are provided.

In 'I/O' mode, there is no Dr.D tape involved. The user program receives from or presents to the Dr.D operation the logical record and accomplishes any other desired processing.

In 'EX' mode, a Dr.D format tape is involved and the user program is presented with the logical record as it is in transit to or from the disk file.

Note: That the user-exits to examine a data tape being saved or restored are not provided. Further, there is no provision for deleting or adding a logical record in 'EX' mode, and no data alteration is possible. These provisions must be made in a program to invoke user-exits:

1. An entry point named 'USEREXIT' must be provided and linkage established to a common area for Dr.D control data. This area consists of 256 bytes in the following format:

The Dr.D parameter card for this operation	-	80	bytes
DTF name	-	7	bytes
Disk file identification field	-	44	bytes
Date (current if 'IO', as saved if 'EX')	-	8	bytes
Julian Day (current if 'IO', as saved if 'Ex')	-	3	bytes
Time (current if 'IO', as saved if 'EX')	-	8	bytes
DLBL type (S=SEQ, D=DAM, C=ISC, E=ISE)	-	1	byte
Error Flag	-	1	byte
Reserved	-	104	bytes

2. A command to Dr.D in the form of a call to send, receive, or look at a record and establish linkage to the data area. 'PUTREC' sends a logical record to Dr.D ('IO' only). 'GETREC' receives a logical record from Dr.D ('IO') or inspects a logical record being processed by Dr.D ('EX').
3. When using 'PUTREC', a signal is required to Dr.D that all is complete. This is accomplished by using 'PUTREC' to send Dr.D the data string 'END OF DATA FILE'.
4. An exit from the user program to Dr.D.

Additional Dr.D parameters and control cards are required:

On the Save/Restore card, 'USER=PHASENAM/MM/TZZZZZ' is present. Phasename is the cataloged name of the user program with provisions as stated above.

'MM' is the mode- 'IO' to Send/Receive logical records, 'EX' to inspect logical records.

'T' is the core image type catalog- 'L' for relocatable or self relocating programs, 'A' for absolute address programs.

'ZZZZZ' is either the decimal length for type 'L' programs, or the hexadecimal load address for absolute address programs.

An 'EXIT' parameter card is also required:

For VSAM files: 'EXITVSAABBBCCCDDDDDEEEEE' (refer to Appendix VII (Summary of Dr.D Commands) for an explanation of the values of AA through EEEEE).

For SAM or DAM files fixed blocked: 'EXITFBFFFFFHHH' where 'FFFFF' is the exact physical blocksize and 'HHH' is the number of logical records per block.

For SAM or DAM files fixed unblocked: 'EXITFUFFFFFGGG' where 'FFFFF' is the same as above and 'GGG' is the key length. If a key is present, it must start in the first position of the logical record.

The 'EXIT' cards for FB/FU are for proper de-blocking and disk writing.

Examples of User-Exit Functions

Example 1. ANS COBOL

```
// JOB MYJOB
// OPTION CATAL
   PHASE MYPROG,*
   INCLUDE GETREC
// EXEC FCOBOL
       IDENTIFICATION DIVISION.
       PROGRAM-ID. MYPROG.
            .
       WORKING-STORAGE SECTION.
       77  DATA-END PIC X(16) VALUE 'END OF DATA FILE'.
       01  MY-RECORD.
            .
            .
       LINKAGE-SECTION.
       01  DOCTOR-AREA.
           05 DOCTOR-CARD            PIC X(80).
           05 DOCTOR-DTF-NAME        PIC X(7).
           05 DOCTOR-DISK-ID         PIC X(44).
           05 DOCTOR-DATE            PIC X(8).
           05 DOCTOR-JULIAN-DAY      PIC 999.
           05 DOCTOR-TIME            PIC X(8).
           05 DOCTOR-DLBL-TYPE       PIC X.
           05 DOCTOR-ERROR-FLAG      PIC X.
           05 FILLER                 PIC X(104).
       PROCEDURE DIVISION.
           ENTRY 'USEREXIT' USING DOCTOR-AREA.
            .
            .
       READ-DOCTOR.
           CALL 'GETREC' USING MY-RECORD.
           IF MY-RECORD = DATA-END
               GO TO END-OF-INPUT.
  (or)
       WRITE-DOCTOR.
           CALL 'PUTREC' USING MY-RECORD.
           IF DOCTOR-ERROR-FLAG = HIGH-VALUE
               GO TO KEY-SEQUENCE-ERROR.
            .
            .
       END-OF-JOB.
            .
       CALL 'PUTREC' USING DATA-END.
       GOBACK.
```

Example 2. Run JCL Example.

This example restores file from Job D3 with user-exit processing.

```
// JOB U15
// UPSI 00100000           LOG FILE COUNTS
// DLBL ISAMFLE,'MASTER',99/365,ISC
// EXTENT SYS001,111111,4,1,40,20
// EXTENT SYS002,222222,1,2,20,3980
// TLBL TAPEIN
// ASSGN SYS008,280
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILE=SAMFLE,FAST=YES,USER=MYPROG/IO/L104000
EXITIS000040090010100001  (USER-EXIT FILE DEFINITION)
** 06/29/74 498  1  1     (USER-EXIT PROGRAM DATA CARD)
/*                        (FOR USER-EXIT PROGRAM)
/*                        (FOR Dr.D )
/&
```

Additional Notes:

If user programs use UPSI switches, the 'UPSI=' must be used on the Dr.D control card to override that in the JCL. 'GETREC' and 'PUTREC' are mutually exclusive.

Card input for user programs must follow immediately after the appropriate 'EXIT' card and must be terminated by a '/*' card in addition to the '/*' required for Dr.D.

Multiple Dr.D functions may be performed in one execution of Dr.D, including more than one user-exit program (or the same user-exit program more than once), with or without card data to one or more of the user-exit programs.

User programs can be any mix of 'IO' or 'EX' modes in any single execution of Dr.D.

In order for user-exit programs to execute properly, 'GETREC' or 'PUTREC' must be the first entry points in the link-edited program. This is most easily accomplished by placing 'INCLUDE PUTREC' or 'INCLUDE GETREC' immediately after the 'PHASE...' card.

The only current use for DOCTOR-ERROR-FLAG is when an ISAM create encounters a key not greater than the previous key written. High-value is returned in the DOCTOR-ERROR-FLAG.

User-exit programs must not use 'STOP RUN' statements, use 'GOBACK' statements instead.

Summary of Control Card Formats for User-Exit Functions

Enable User-Written Programs

To enable user-written programs to access, modify, or delete logical records during backup and restore.

```
RESTORE=DATA(,...ETC...),USEREXIT=NF/PPPPPPPP
```

To enable user-written programs to generate intercept or inspect logical records during Dr.D runs

```
RESTORE=DATA(,...ETC...),USER=PHASENAM/MM/TZZZZZZ
EXITISAABBBCCCDDDDDEEEEE
EXITFUFFFFFGGG
EXITFBFFFFFHHH
```

VSE-SP Library Functions

Saving, Restoring, Copying VSE-SP Libraries

Dr.D can save/restore/copy all/selected members of a VSE-SP library. Member selection is based on sub-library, type, member name, and time stamp. Generic groups of members can be selected by use of the ('/') character. Sub-library and type selectors are prefixed by the plus ('+') and exclamation ('!'). Members of one sub-library may be restored to another sub-library by use of the 'TO' sub-library designation. This is accomplished by prefixing the sub-library name with the ampersand ('&'). If the 'TO' sub-library does not exist in the receiving library, Dr.D will create it before adding the selected members.

Restore Method

Dr.D uses two different methods for restoring library members. The assumed method ('LIB=PHYSICAL') uses Dr.D's own member and directory output routines. If 'LIB=LIBR' is specified, then Dr.D invokes the LIBR functions for member and directory output. The two methods differ in several aspects: 1) The LIBR method requires more GETVIS memory and more free space in the library, but may be suitable for libraries in active use. 2) The LIBR method changes the date of the member while 'LIB=P' retains the date as saved.

Timestamp Selection and Incremental Backup

Use of the 'TSTAMP=' keyword allows Dr.D to do incremental backup of libraries to greatly reduce backup time for daily backup. This is accomplished by doing full backups periodically (weekly, monthly etc.) and then doing incremental backups more frequently (daily, twice daily, etc.). The incremental backup would use a time stamp selector that would cause all members changed since the last full backup to be saved each time the backup is run. Recovery of the library then becomes a matter of restoring the last full backup and then restoring the last incremental backup.

Specifying the Library

The assumed 'DLBL' name is 'SYSLIB', but may be changed by use of the 'SF' and 'RF' parameters. Dr.D does not distinguish between VSE-SP system and private libraries, and label information is always required.

Controlling Print Output

Listing of sub-library and member names can be controlled by use of the 'FORMAT=' keyword as follows:

	MEMBER NAMES PRINT		SUB-LIBRARY NAMES PRINT	
	ON LOG	ON LIST	ON LOG	ON LIST
FORMAT=A	YES	YES	YES	YES
FORMAT=B	NO	YES	YES	YES
FORMAT=C	NO	YES	NO	YES
FORMAT=D	YES	NO	YES	NO
FORMAT=E	NO	NO	NO	NO

Examples of VSE-SP Library Functions

Example 1. Save a library.

This example saves an entire library.

```
// JOB SAVE A VSE-SP LIBRARY
// ASSGN SYS007,180
// TLBL TAPEOUT
// DLBL SYSLIB,'VSE-SP LIBRARY'
// EXTENT ,FBA250
// EXEC DRD,SIZE=200K
   SAVE=LIBRARY,UPSI=111,PRINT=BOTH
   SAVE=NOP,TAPE=REW
/*
/&
```

Example 2. Restore a library.

Example 2 restores an entire library.

```
// JOB RESTORE A VSE-SP LIBRARY
// ASSGN SYS008,180
// TLBL TAPEIN
// DLBL SYSLIB,'VSE-SP LIBRARY'
// EXTENT ,FBA250
// EXEC DRD,SIZE=200K
   RESTORE=LIBRARY,UPSI=111
   RESTORE=NOP,TAPE=RUN
/*
/&
```

Example 3. Save selected Members.

This example saves selected members of a VSE-SP library, but only members changed in the preceding 7 days.

```
// JOB SAVE SELECTED MEMBERS OF A VSE-SP LIBRARY
// TLBL TAPEOUT
// DLBL SYSLIB,'VSE-SP LIBRARY'
// EXTENT SYS001
// EXEC DRD,SIZE=200K
   SAVE=LIBRARY,UPSI=111,MEMBER=SELECT,TSTAMP=>-6
   +PR$260               SUB-LIBRARY SELECTOR
   !PHASE                MEMBER TYPE SELECTOR
   ////////              SELECT ALL MEMBERS OF PR$260.PHASE
   !E                    SELECT ANOTHER TYPE
   DTF/////              SELECT DTF'S IN PR$260.E
   /@                    END OF SELECTORS
   SAVE=NOP,TAPE=REW
/*
/&
```

Example 4. Restore selected Members.

This example restores selected members to a new sub-library.

```
// JOB RESTORE SELECTED MEMBERS
// TLBL TAPEIN
// DLBL SYSLIB,'VSE-SP LIBRARY'
// EXTENT SYS001
// EXEC DRD,SIZE=200K
   RESTORE=LIBRARY,UPSI=111,MEMBER=SELECT
   +PR$260                  SUB-LIBRARY PR$260
   &PR$261                  RESTORE-TO PR$261
   !PHASE                   SELECT TYPE 'PHASE'
   IJZ/////                 SELECT MEMBERS PREFIXED BY 'IJZ'
   /@
   RESTORE=NOP,TAPE=RUN
/*
/&
```

Both 'LIB=NEW' and 'LIB=ADD' are supported. 'LIB=ADD' is assumed with restored members always re-using their space if they are in the receiving library.

If the 'LIB=OLD' function is needed, either allow 'LIB=NEW' to empty the library or use the IBM LIBR program to delete the desired library/sub-library member. In any case, if 'LIB=OLD' is specified, 'LIB=ADD' will be used instead.

If 'PRINT=YES' is specified along with UPSI XX1, a list of all members saved, restored, or copied is produced on SYSLST. If 'PRINT=BOTH' and UPSI XX1 are specified the member list is produced on SYSLOG and SYSLST.

Library Space Considerations

In estimating library space requirements, a good rule of thumb is about 10% of the library will be needed for directory and high level indexes. If the library fills up during RESTORE/COPY, the members up to the one noted in message 74 are restored and the library is intact (use the 'LIB=EXTEND' feature if you wish to enlarge the library). When determining whether a member will fit, an estimate is made of the number of additional directory blocks and higher level indexes needed. This estimate insures the successful completion of the RESTORE/COPY but may allow one or two more blocks than are actually needed for the operation. If library space is limited, always issue a LIBR 'RELEASE' command before attempting the restore.

Library Create/Extend

Dr.D can add extents to libraries in either VSAM or NON-VSAM space, but VSE-SP does not allow the SYSRES library to have more than one extent. These functions are independent of any other library function, therefore a follow-on restore/copy operation is required if members are to be added to the library. If the library is in NON-VSAM space, Dr.D will issue an open for each JCL EXTENT card presented, with the corresponding VSE 'EQUAL FILE ID IN VTOC' message, even if the library is being extended.

Note: That the 'RFILENAME=' parameter is required; RF=SYSLIB is NOT assumed.

The operator should respond with 'DELETE' after verifying that the correct file ID is listed. If the library is in VSAM space and 'LIBRARY=CREATE' is specified, a single open is issued to allocate the first extent. If 'LIBRARY=EXTEND' is specified, the library is checked to determine the number of existing extents before opening as output to allocate a new extent.

Example 5. Create Non-VSAM library.

This example creates a NON-VSAM Library.

```
// JOB CREATE NON-VSAM
// DLBL SYSLIB,'VSE-SP NON-VSAM LIBRARY'
// EXTENT SYS001,222222,1,0,15,1500
// EXEC DRD,SIZE=200K
   RESTORE=LIBRARY,UPSI=111,LIBRARY=CREATE,RFILENAME=SYSLIB
/*
/&
```

Example 6. Extend a Non-VSAM library.

This job stream extends a non-VSAM library.

```
// JOB EXTEND NON-VSAM
// DLBL SYSLIB,'VSE-SP NON-VSAM LIBRARY'
// EXTENT SYS001,222222,1,0,15,1500
// EXTENT SYS001,222222,1,1,6000,750 (extent to be added)
// EXEC DRD,SIZE=200K
   RESTORE=LIBRARY,UPSI=111,LIBRARY=EXTEND,RFILE=SYSLIB
/*
/&
```

Example 7. Create a VSAM library.

This example creates a VSAM library.

```
// JOB CREATE VSAM
// DLBL SYSLIB,'VSE-SP VSAM LIBRARY',,VSAM,CAT=IJSYSUC,DISP=(OLD,KEEP)
// EXEC DRD,SIZE=200K
   RESTORE=LIBRARY,UPSI=111,LIBRARY=CREATE,RFILE=SYSLIB
/*
/&
```

Example 8. Extend a VSAM library.

This example extends a VSAM library.

```
// JOB EXTEND VSAM
// DLBL SYSLIB,'VSE-SP VSAM LIBRARY',,VSAM,CAT=IJSYSUC,DISP=(OLD,KEEP)
// EXEC DRD,SIZE=200K
   RESTORE=LIBRARY,UPSI=111,LIBRARY=EXTEND,RFILE=SYSLIB
/*
/&
```

Summary of Control Card Format for VSE-SP Libraries

Save a VSE-SP Library

```
SAVE=LIBRARY,SFILENAME=XXXXXXXX(,MEMBER=SELECT/REJECT,TSTAMP=XXXXX
```

Restore a VSE-SP Library

```
RESTORE=LIBRARY,RFILENAME=XXXXXX(,MEMBER=SELECT/REJECT,TSTAMP=XXXXX
FIND=XXXXXX/SKIP=XXX,LIBRARY=CREATE/EXTEND)
```

VSE-SP List Directory Function

The Dr.D list directory function (FUNCTION=LDIR) provides a method for producing a compact (two up format) directory listing with a high degree of selectivity. Members can be selected for listing by their library, sub-library, type, specific or generic member name, and by the 'last update' time stamp. A library recap is produced giving statistics by library, sub-library, and member type.

The report format is mostly self-explanatory. The report heading shows the 'time stamp' used to select the members for listing, the library file ID, the size of the library in blocks, and the number of free blocks and %. For each member is listed its date and time created and last changed (military time), its blocks used and contiguous, and its size. For PHASE member type, the size is the size in bytes while for other types the size is the number of logical records used by the member. Two library recaps are produced (unless suppressed), one by sub-library and another by overall library. Both recaps show by member type the number of members, number of blocks and % library share. And, finally, a library efficiency rating is listed to give an estimate of library disorganization on a scale of 1-100 with 100 indicating the highest efficiency and least disorganization. For information on how to use the 'FORMAT=' parameter please refer to the keyword section of this manual.

Example 1. Using the LDIRECTORY function list all members in Lib. PRD2.

List all members of the PRD2 library.

```
// JOB LIST ALL MEMBERS
// EXEC DRD,SIZE=200K
   FUNCTION=LDIRECTORY,SFILE=PRD2
/*
/&
```

Example 2. List members changed or added to the library.

List all members changed / added to the library in the last two days.

```
// JOB LIST MEMBERS ADDED/CHANGED
// EXEC DRD,SIZE=200K
   FUNCTION=LDIRECTORY,SFILE=PRD2,TSTAMP=>-1
/*
/&
```

Example 3. List all phases in the library regardless of sub-library.

List all members of type 'PHASE' regardless of their sub-library.

```
// JOB LIST MEMBER TYPE PHASE
// EXEC DRD,SIZE=200K
   FUNCTION=LDIRECTORY,SFILE=PRD2,MEMBER=SELECT
   +////////                          SELECT ALL SUB-LIBRARIES
   !PHASE                             SELECT PHASES
   ////////                           SELECT ALL MEMBERS
/@                                    TERMINATE SELECTION
/*
/&
```

Example 4. List all phases that have changed in sub-library 'TEST'.

List all PHASE type members changed since MAY 1,1991 AT 9 AM, but only those in sub-library 'TEST'.

```
// JOB LIST MEMBER TYPE PHASE
// EXEC DRD,SIZE=200K
   FUNCTION=LDIRECTORY,SFILE=PRD2,MEM=SELECT,TSTAMP=91121.0900
   +TEST                              SELECT SUB-LIBRARY 'TEST'
   !PHASE                             SELECT PHASES
   ////////                           SELECT ALL MEMBERS
/@                                    TERMINATE SELECTION
/*
/&
```

Example 5. List library statistics without member listing.

List just library statistics only, with no member printout.

```
// JOB LIST LIBRARY STATS ONLY
// EXEC DRD,SIZE=200K
   FUNCTION=LDIRECTORY,SFILE=PRD2,FORMAT=B
/*
/&
```

Example 6. List all members in sub-library 'TEST' whose names begin with 'DRD' regardless of type.

List all members in sub-library 'TEST' whose names begin with 'DRD' regardless of their type. Suppress the library statistics recap.

```
// JOB LIST SELECTED MEMBERS WITHOUT LIBRARY RECAP STATISTICS
// EXEC DRD,SIZE=200K
   FUNCTION=LDIRECTORY,SFILE=PRD2,MEMBER=SELECT,FORMAT=A
   +TEST                            SELECT SUB-LIBRARY 'TEST'
   !////////                        SELECT ALL TYPES
   DRD/////                         SELECT ALL 'DRD' MEMBERS
/@                                TERMINATE SELECTION
/*
/&
```

Miscellaneous Functions

Creating a Format-1 Label

Dr.D creates an SD or DA file label by specifying 'RE=PU' (put Format-1) and 'RF=' (specify filename).

Sometimes you know you have valid data on a pack but you can't access it because the Format-1 Label isn't there. It could have been accidentally deleted or the File-ID incorrectly specified, or it may have never existed.

Example 1. Put Format-1 Label.

Assuming a corrupted label for a random master file, this example recreates it.

```
// JOB P1
// DLBL RANMAS,'MASTER FILE',99/360,DA
// EXTENT SYS001,777771,1,0,20,3980
// EXTENT SYS002,777772,1,1,20,3980
// ASSGN SYS001,131
// ASSGN SYS002,132
// EXEC DRD,SIZE=200K
RESTORE=PUTLABEL,RFILENAME=RANMAS
/*
/&
```

Summary of Control Card Format for Put Format-1 Label

Put a Format-1 Label:

```
RESTORE=PUTLABEL,RFILENAME=XXXXXXX
```

Erasing a Disk

Sometimes it would be nice to be able to clear or erase a portion of a disk.

Specifying 'RESTORE=ERASE' will accomplish this goal. Standard CKD R0 records are written and the tracks are erased; FBA blocks are cleared.

The disk is assumed to be at SYS002, but this may be overridden by entering 'RSYSNO=SYSXXX' or 'RVOLID=SSSSSS' (SSSSSS=VOLSER). The area of the disk to be erased/cleared is defined by use of the 'RBEGIN' and 'RHALT' parameters. These parameters are one to eight numeric digits in length. For CKD disk types, the two digits are assumed to be the head number and those to the left are the cylinder (CCCCCCHH). For FBA disk types, the whole number is an FBA relative block number. In any case, no leading zeros are needed. If no ending ('RHALT') disk address is entered, the ending is made equal to the beginning so that only one track/FBA block is erased/cleared.

Example 2. Erase Disk.

Assume two 3380's on 131 and 132. This example erases all of the pack on 131 and the last half of the pack on 132. It also erases cylinder 705 of a 3375 on 155.

```
// JOB E2
// ASSGN SYS002,131
// ASSGN SYS003,132
// ASSGN SYS004,155
// EXEC DRD,SIZE=200K
RESTORE=ERASE,RBEGIN=101,RHALT=84500
RESTORE=ERASE,RBEGIN=40000,RHALT=84500,RSYSNO=SYS003
RESTORE=ERASE,RBEGIN=70500,RHALT=70518,RSYSNO=SYS004
/*
/&
```

Summary of Control Card Format for Erasing the Disk

Erase Disk:

CKD→

```
RE=ERASE(,RSYSNO=SYSXXX)(,RBEGIN=CCCHH)(,RHALT=CCCHH)(,MOD11=YES)
```

FBA→

```
RE=ERASE(,RSYSNO=SYSXXX)(,RBEGIN=BBBBBB)(,RHALT=BBBBBB)
```

Default:

RSYSNO=SYS002

Altering the Access, Creation, or Expiration date of a File.

Using the 'RESTORE=EXPIRDATE' facility of Dr.D does not really restore anything, but does update the expiration date of a file on a pack. This is the assumed default value if not specified on the control card.

The correct file is communicated to Dr.D via the 1-44 alphanumeric 'RIDENT=' file identification parameter or by use of the 'RF=' parameter.

Note: If an operand has embedded blanks, it must be enclosed in quotes (').

The disk is assumed to be at SYS002, but can be altered by specifying the 'RSYSNO=' or 'RVOLSER=' parameter.

The desired expiration date is specified by the 'DATE=NNNNN' parameter. 'NNNNN' is a 5-digit number representing the Julian date in the form of 'YYDDD'. A file may be retained indefinitely by specifying 'DATE=RETAIN' (equivalent to 'DATE=99367). A file may be deleted by putting 'DATE=DELETE' (equivalent to 'DATE=00001').

Example 12. Alter expiration date.

Assume a file on 131 whose file ID is 'TEST FILE' which expires today (January 1, 1990). This example alters the expiration date to a week from today.

```
// JOB E12
// ASSGN SYS002,131
// EXEC DRD,SIZE=200K
RESTORE=EXPIRATIONDATE,RIDENT='TEST FILE',DATE=90008
/*
/&
```

This function can also be used to set the time stamp and flags in a VTOC format one label to the values needed for a VSAM data space. This might be needed if the data space was saved and then restored as a direct access file.

Example 12a. Change restored VSAM space from direct access to VSAM.

```
// JOB E12A
// ASSGN SYS002,131
// DLBL VSPACE,'Z9999992.VSAMDSPC.TA593912.T437E5BA'
// EXEC DRD,SIZE=200K
   RE=EX,RFILE=VSPACE,DATE=RETAIN,VSAM=SET,TS=80000000
/*
/&
```

Note that the time stamp is entered as hex code.

Example 13. Selective Save.

Assume a 3375 SYSRES with three files that never expire: SYSTEM residence, SYSTEM recorder, and POWER/VS queue. This example executes a 'SAVE=CUR' on the SYSRES pack but ignores the POWER/VS queue.

```
// JOB E13
// ASSGN SYS007,280
// TLBL TAPEOUT,'SYSRES SAVE'
// EXEC DRD,SIZE=200K
RESTORE=EXPIRED,RIDENT='POWER/VS QUEUE',RSYSNO=SYSRES,DATE=DELETE
SAVE=IPL
SAVE=CURRENT,SSYSNO=SYSRES
RESTORE=EXPIRDATE,RIDENT='POWER/VS QUEUE',RSYSNO=SYSRES,DATE=RETAIN
/*
/&
```

Summary of Control Card Format for Alter Expiration Date

Alter Expiration Date:

```
RESTORE=EXPIRDATE,RIDENT=XXX...XXX(,RSYSNO=SYSXXX),DATE=YYDDD
                                                   DATE=DELETE
                                                   DATE=RETAIN
```

Default:

RSYSNO=SYS002

Miscellaneous Operations

This section discusses the following operands:

- Posting a message to the operator.
- Specifying a control card count.
- Altering the blocksize.
- Specifying tape compression
- Specifying Disk Volume Serial Numbers

Posting a Message to the Operator

Specifying 'OP=SOME MESSAGE' causes Dr.D to log out an operator message on SYSLOG. This operand may appear on a control card in conjunction with a 'SAVE/RESTORE/COPY=' function or all by itself. If any Dr.D operand contains embedded blanks it must be enclosed in quotes.

Wrong: OP=MOUNT pack 222222 on 131
Right: OP='MOUNT pack 222222 on 131'

To cancel Dr.D at this point, type 'C'. To continue, perform an 'EOB' operation.

Example 1. Post a Message.

This example posts an informational message telling the operator to save SYSRES only after all links have been done. Two alternate job streams for achieving this goal follow:

```
// JOB O1 ONE WAY
// ASSGN SYS007,280
// TLBL TAPEOUT,'SYSRES SAVE'
// EXEC DRD,SIZE=200K
   SA=NOP,OPERATOR='SAVE SYSRES IF LINKS ARE DONE'
   SAVE=IPL
   SAVE=CURRENT,SSYSNO=SYSRES
/*
/&

// JOB O1 ANOTHER WAY
// TLBL TAPEOUT,'SYSRES SAVE'
// ASSGN SYS007,280
// EXEC DRD,SIZE=200K
SAVE=IPL,OPERATOR='SAVE SYSRES IF LINKS ARE DONE'
SAVE=CURRENT,SSYSNO=SYSRES
/*
/&
```

Specifying a Control Card Count

To prevent accidental loss or insertion of control cards into the JCL, a card count feature is provided. All cards that have 'SAVE/RESTORE/COPY=' on them are countable cards. Not included in the count are their logical continuation cards if they have any '$$$$$$' cards or the '/*'. The card count is communicated to Dr.D by the operand 'CARDCOUNT=NN'. It may appear anywhere in the sequence of control cards, but may only appear once. The count starts with the card on which 'CA=NN' appears and ends with the last one. The following are some correct examples of saving five data files.

```
            A                          B
// EXEC DRD                  // EXEC DRD
SA=DA,SF=FIL01,CA=05         SA=DA,SF=FIL01
SA=DA,SF=FIL02               SA=DA,SF=FIL02
SA=DA,SF=FIL03               SA=DA,SF=FIL03
SA=DA,SF=FIL04               SA=DA,SF=FIL04,CA=02
SA=DA,SF=FIL05               SA=DA,SF=FIL05
/*                           /*
```

In example A, the operator would be informed if any card was missing or added.

In example B, the same holds true with the last two cards. Insertions or deletions before card number four would go undetected.

Altering the Blocksize

`BLOCKSIZE=`

The assumed tape blocksize is 65K (double buffered), but may be altered by specifying 'BLOCKSIZE=NNNNN' on the control card, where 'NNNNN' is a 5-digit number less than 65,496 and equal to or greater than the largest physical disk block +16, to be saved/copied.

Once specified, the blocksize is in effect for the remainder of the program's execution, or until another 'BLOCKSIZE=' is submitted. The blocksize may be increased in order to get more data on a tape. However, if the tape is read back in during a restore, a 'BLOCKSIZE=' must be put on the tape so Dr.D can accommodate the bigger blocksize.

When used with a data file save and 'DT=YES', NNNNN specifies the blocksize of the tape record. It must be less than 65496 and at least as large as the logical record length plus 10. In no event will other than full logical records be written to tape.

When used during copy operations, memory requirements may be reduced or increased with a resulting change in performance. For VSE machines with limited memory, performance may actually be improved by a reduction in blocksize through elimination of excessive paging. With adequate real memory, a larger block size should prove beneficial. Do not use this parameter on the same

control card as 'FIND', 'SKIP' or 'POSITION'. Instead use a preceding 'RE=NOP' control card to set the desired blocksize.

Specifying Tape Compression

CP=YES/NO

When saving to tape, 'CP=YES' causes tape output to be compressed, thereby reducing tape usage and channel access, with improvement in backup speed (provided CPU time is adequate). Once invoked, it remains in effect until 'CP=NO' is specified, or execution ends.

Example 3. Specify Tape Compression.

This example describes a specific tape compression.

```
// JOB O3
// ASSGN SYS007,280
// TLBL TAPEOUT,'RELO'
// EXEC DRD,SIZE=200K
SAVE=PCORE,BLOCKSIZE=65496,CPRESS=YES
SAVE=PRELO,CPRESS=NO
/*
/&
```

Specifying Disk Volume Serial Numbers

SVOLSER,RVOLSER,CVOLSER=

The 'SV/RV/CV' operands are used to specify the disk volume serial number(s) for operations where normal LIOCS processing is not used. Whenever the operand is entered, Dr.D checks to see if there is a usable assignment for the disk(s), and attempts to make an assignment if necessary, thereby eliminating the need for JCL assign statements. If a proper assignment cannot be made, or the disk volser specified cannot be located, an error message is produced. Dr.D will attempt to make necessary disk assignments anytime the disk volser is known from the 'SV/RV/CV, or EXTENT statement'.

Example 4. Specify disk Volume Serial number.

This example is for the operator that needs to save a pack whose serial number is 654321, but is unsure of the physical disk address.

```
// JOB O4 SAVE DISK VOLSER 654321
// ASSGN SYS007,280
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
SAVE=ALL,SVOLSER=654321
/*
/&
```

Example 5. Copy entire Cylinder.

This example copies the entire second cylinder from 3380 on 131 (serial ABCDEF) to 3380 on 132 (serial 123456).

```
// JOB O5 FIRST
// EXEC DRD,SIZE=200K
COPY=PHYS,CBEGIN=100,CHALT=114,RMODE=IMAGE,RVOLSER=123456,SV=ABCDEF
/*
/&
```

Summary of Control Card Formats for Miscellaneous Operands

Log Message on Console:

```
OPERATOR=XXX...XXX
FUNCTION:  LOG MSG PRIOR TO SAVE/RESTORE/COPY

SAVE/RESTORE/COPY=XX,(,...ETC...),OPERATOR=XXX...XXX
```

Specify Control Card Count:

```
SAVE/RESTORE/COPY=XX,(,...ETC...),CARDCOUNT=NN
```

Alter Blocksize:

```
SAVE/RESTORE/COPY=XX,(,...ETC...),BLOCKSIZE=NNNNN
```

Invoke Compression on Tape:

```
SAVE=SOMETHING(,...ETC...),CPRESS=YES
```

Specify Input Disk Volume Serial Number:

```
SAVE/COPY=XX,(,...ETC...),SVOLSER=XXXXXX
```

Specify Output Disk Volume Serial Number:

```
RESTORE/COPY=XX,(,...ETC...),RVOLSER=XXXXXX
```

Working With Tape Files

Summary of Control Card Formats for Tape Functions

Set Tape DDNAME To Other Than Assumed:

```
SAVE/RESTORE=SOMETHING(,...ETC...),TDDNAME=DDDDDDD
```

Set Tape SYSNOS To Other Than Assumed:

```
SA/RE=SOMETHING(,...ETC...),TAPEOUT=SYSXXX,TAPETWO=SYSYYY,
TAPEIN=SYSZZZ
```

Check Control Card Sequence:

```
SAVE=SOMETHING(,...ETC...),SEQUENCE=NN
RESTORE=SOMETHING
COPY=SOMETHING
```

Close Output Tape Prior To Saving:

```
SAVE=SOMETHING(,...ETC...),TAPE=REW
                           TAPE=RUN
                           TAPE=NOREWIND
```

Close Input Tape Prior to Restoring:

```
RESTORE=SOMETHING(,...ETC...),TAPE=REWIND
                              TAPE=RUN
                              TAPE=NOREWIND
```

Position Input Tape Prior to Restoring:

```
RESTORE=SOMETHING(,...ETC...),SKIPFILES=NNN
                               FINDFILE=XXXXXXX
```

In some cases, users may wish to use tape 'SYSNOS or DDNAMES' (TAPEIN/TAPEOUT) other than the assumed values. To do so, add the 'TO=(TAPEOUT=)', and/or 'TT=(TAPETWO=)', and/or 'TI=(TAPEIN=)' keywords to the control card.

Note: If the first is 'SAVE=IP', you must repeat on the next card, as well.

Use the operand 'TD=DDDDDDD' to change the tape 'DDNAME' from TAPEIN/TAPEOUT. TAPEIN is changed if doing a restore; TAPEOUT if doing a save. Except for the case of SA=IP, the values entered will hold throughout an execution of Dr.D.

You can do any combination of Saves, Restores, and Copies within one execution of Dr.D. All Dr.D save/restore tapes have standard labels, unless 'LABEL=NO' is specified. To illustrate, consider a user has saved three data files in a row:

```
// EXEC DRD,SIZE=200K
SAVE=DATA,SFILE=FILEX
SAVE=DATA,SFILE=FILEY
SAVE=DATA,SFILE=FILEZ
/*
```

The save tape would have the following format:

```
VOL1 LABEL    (DOS)
HDR1 LABEL    (DOS)
TAPEMARK
Dr.D  HDR REC FOR FILE X SEQUENCE NUMBER IS 001
Dr.D  DELIMITER
   DATA RECS FOR FILE X
Dr.D  DELIMITER
Dr.D  EOF REC FOR FILE X
Dr.D  DELIMITER
Dr.D  HDR REC FOR FILE Y SEQUENCE NUMBER IS 002
Dr.D  DELIMITER
   DATA RECS FOR FILE Y
Dr.D  DELIMITER
Dr.D  EOF FOR FILE Y
Dr.D  DELIMITER
Dr.D  HDR REC FOR FILE Z SEQUENCE NUMBER IS 003
Dr.D  DELIMITER
   DATA RECS FOR FILE Z
Dr.D  DELIMITER
Dr.D  EOF REC FOR FILE Z
Dr.D  DELIMITER
TAPEMARK
EOF1 LABEL    (DOS)
TAPEMARK
```

The control stream for restore would appear as follows:

```
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILE=FILEX
RESTORE=DATA,RFILE=FILEY
RESTORE=DATA,RFILE=FILEZ
/*
```

It is very important to keep the save and restore control cards in the same sequence. In order to prevent accidental permutation of the control cards, use the 'SEQUENCE=NN' operand. On the save side the sequence number (if specified)

is checked against the one internally generated for the Dr.D HDR record. A no match indicates an error.

```
// EXEC DRD,SIZE=200K
SAVE=DATA,SFILE=FILEX,SEQUENCE=01,CARDCOUNT=03
SAVE=DATA,SFILE=FILEZ,SEQUENCE=03 <---ERROR INDICATED
SAVE=DATA,SFILE=FILEY,SEQUENCE=02
/*
```

Likewise, on the restore side, the sequence number (if specified) is checked against the one in the Dr.D HDR record.

```
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILE=FILEX,SEQUENCE=01,CARDCOUNT=03
RESTORE=DATA,RFILE=FILEZ,SEQUENCE=03 <---ERROR INDICATED
RESTORE=DATA,RFILE=FILEY,SEQUENCE=02
/*
```

While many different types of saves may be performed during the execution of Dr.D, they are all embedded within one DOS tape file.

The use of 'SKIP=NNN' or 'FIND=XXXXXX' allows users to specify which file within a Dr.D save (single execution) with which to begin restoring. This operation allows a user to restore selective files from the save tape.

The SKIP command starts with the first file saved and reads until the skip 'NNN' is satisfied. The 'FIND=' command searches the Dr.D file until either the end of file is reached, or the find search argument matches the tape.

The search argument may be the DLBL name used during the save, the file ID, or the volume serial number of the saved disk/disk segment. If 'FIND=XXXXXXX' is specified, the DLBL name is the search argument. If the object of the 'FIND=' is enclosed in quotes ('), the search argument is the 44 character file ID field.

If 'FIND=(SSSSSS)' is specified, the volume serial of the saved disk/disk segment is searched. For multivolume tape files, 'SKIP' requires the tape volumes to be mounted in order, whereas 'FIND' allows searching to begin with any volume. If 'FIND' is used beginning with other than the first volume, the user must keep track of which files are on which volume. See 'DOCTOR 616' in the message section for additional tape positioning.

Example 1. Restore of First and Third files using the FIND parameter.

In this example, 'FIND=SSSSSS' is specified and is restoring the first and third files only.

```
// JOB RESTORE FIRST AND THIRD USING FIND=
// ASSGN SYS025,280
// TLBL TAPEIN,'SOME SAVES'
// EXEC DRD,SIZE=200K
RESTORE=NOP,TINPUT=SYS025
RESTORE=DATA,RFILE=FILEX,CARDCOUNT=2,FIND=FILEX
RESTORE=DATA,RFILE=FILEZ,FIND='FILE ID OF 3RD FILE'
/*
/&
```

Example 2. SKIP before restoring.

In this example, 'SKIP=NNN' will skip over NNN saved items on the tape before restoring.

```
// JOB RESTORE FIRST AND THIRD
// ASSGN SYS008,280
// TLBL TAPEIN,'SOME SAVES'
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILE=FILEX,CARDCOUNT=2
RESTORE=DATA,RFILE=FILEZ,SKIP=001
/*
/&
```

Example 3. Stack multivolume files.

An alternative method is to allow DOS to do the positioning. Dr.D opens and closes both input and output with 'NO REWIND'. Therefore you may stack multivolume files with multiple executions of Dr.D.

```
// JOB SAVE THREE FILES
// ASSGN SYS007,280
// TLBL TAPEOUT,'FILEX SAVE',,,,1
// EXEC DRD,SIZE=200K
SAVE=DATA,SFILENAME=FILEX
/*
// TLBL TAPEOUT,'FILEY SAVE',,,,2
// EXEC DRD,SIZE=200K
SAVE=DATA,SFILE=FILEY
/*
// TLBL TAPEOUT,'FILEZ SAVE',,,,3
// EXEC DRD,SIZE=200K
SAVE=DATA,SFILE=FILEZ
/*
/&
```

Example 4. Restore of multivolume file tape created in Example 3.

This examples restores all three files from example 3.

```
// ASSGN SYS008,280
// TLBL TAPEIN,'FILEX SAVE'
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILE=FILEX
/*
// TLBL TAPEIN,'FILEY SAVE'
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILE=FILEY
/*
// TLBL TAPEIN,'FILEZ SAVE'
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILE=FILEZ
/*
/&
```

Example 5. Restore First and Third files of a multivolume file backup.

This job stream restores only the first and third files.

```
// JOB RESTORE STACKED FIRST AND THIRD
// ASSGN SYS008,280
// TLBL TAPEIN,'FILEX SAVE',,,,1
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILE=FILEX
/*
// TLBL TAPEIN,'FILEZ SAVE',,,,3
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILE=FILEZ
/*
/&
```

VSE will automatically position the tape if it is at load point. To restore just the second file:

```
// JOB RESTORE SECOND FILE
// ASSGN SYS008,280
// TLBL TAPEIN,'FILEY SAVE',,,,2
// EXEC DRD,SIZE=200K
RESTORE=DATA,RFILE=FILEY
/*
/&
```

To further facilitate tape handling you may specify 'TAPE=XX'. Dr.D will not open the tapeout file until it hits a 'SAVE='. Likewise, Dr.D will not open the 'TAPEIN' file until it hits a 'RESTORE='. When Dr.D reads a '/*' denoting the end of control cards, it will close any open tape files before terminating. You may override the normal sequence of events by specifying:

```
TAPE=REW
```

```
TAPE=RUN
TAPE=NOR
```

in conjunction with 'SAVE=' or 'RESTORE='. If specified with 'SAVE=', the tapeout tape will be closed with the appropriate REWIND, REWIND and UNLOAD, or NO REWIND, and reopened before saving. Likewise, specifying 'TAPE=' on a 'RESTORE=' control card will close the tapein tape with the specified option and reopen the tape prior to restoring.

Example 6. Save two packs.

This job stream performs two backups and closes the tape out, informing the operator to mount the next tape.

```
// JOB SAVE TWO PACKS
// ASSGN SYS003,131
// ASSGN SYS004,132
// ASSGN SYS010,280
// TLBL TAPEOUT,'SAVE'
// EXEC DRD,SIZE=200K
SAVE=ALL,SSYSNO=SYS003,TOUT=SYS010
SAVE=ALL,SSYSNO=SYS004,TAPE=RUN,OPERATOR='MOUNT NEXT TAPE'
/*
/&
```

It is important to remember that if 'TA=XXX' is specified on a control card, the tape is closed immediately with the appropriate option before anything is saved or restored. By putting 'TA=XXX' on the next card, the user can close a tape after the save or restore.

This raises some questions. In the cases where there is no next card or the user wants to rewind the tape after a save but the next card is a restore card, a dummy function 'NOP' is employed.

Specifying 'RE=NO' or 'SA=NO' will put the program in the restore or save mode, but does not restore or save anything. It is intended to be used in conjunction with 'TA=XXX' only.

Example 7. Save and unload tape.

These examples demonstrate a correct and incorrect approach to saving a disk on 131 to a tape on 280 and then unloading the tape.

```
Correct:      // JOB RUN USED RIGHT
              // TLBL TAPEOUT
              // ASSGN SYS001,131
              // ASSGN SYS007,280
              // EXEC DRD,SIZE=200K
              SAVE=ALL
              SAVE=NOP,TAPE=RUN
              /*
              /&
```

```
Incorrect:    // JOB RUN USED WRONG
              // TLBL TAPEOUT
              // ASSGN SYS001,131
              // ASSGN SYS007,280
              // EXEC DRD,SIZE=200K
              SAVE=ALL,TAPE=RUN
              /*
              /&
```

Example 8. Copy using intermediary scratch tape.

Copy a disk on 131 to one on 132 and a disk on 133 to one on 134. Don't go disk to disk, instead use a scratch tape as an intermediary.

```
// JOB DISK TO TAPE TO DISK
// TLBL TAPEOUT,'SCRATCH'
// TLBL TAPEIN,'SCRATCH'
// ASSGN SYS010,131
// ASSGN SYS011,132
// ASSGN SYS012,133
// ASSGN SYS013,134
// ASSGN SYS007,280
// ASSGN SYS008,280
// EXEC DRD,SIZE=200K
SAVE=ALL,SSYSNO=SYS010
SAVE=NOP,TAPE=REW
RESTORE=ALL,RSYSNO=SYS011
RESTORE=NOP,TAPE=REW
SAVE=ALL,SSYSNO=SYS012
SAVE=NOP,TAPE=REW
RESTORE=ALL,RSYSNO=SYS013
/*
/&
```

If multiple volume backups are made on the same tape or set of tapes, the tape may be positioned for restore by use of the 'SKIP' or 'FIND' parameters.

If 'SKIP=NNN' is used, positioning must begin with the first tape of the set, but if 'FIND=(SSSSSS)' is used, positioning may begin with any volume, as long as the user mounts a volume containing the desired backup, or one preceding it.

Example 9. Save multiple volumes.

This example demonstrates the use of multiple volume backups on the same tape.

```
// JOB SAVE MULTIPLE VOLUMES
// TLBL TAPEOUT
// ASSGN SYS007,280
// EXEC DRD,SIZE=200K
SAVE=ALL,SVOLSER=111111
SAVE=ALL,SVOLSER=222222
SAVE=ALL,SVOLSER=333333
SAVE=NOP,TAPE=REW
/*
/&
```

Example 10. Restore from multiple volume save.

This example demonstrates how to position the tape when you have multiple volume backups on the same tape.

```
// JOB RESTORE FROM MULTIPLE VOLUME SAVE
// TLBL TAPEIN
// ASSGN SYS008,280
// EXEC DRD,SIZE=200K
RESTORE=ALL,RVOLSER=222222,FIND=(222222)
RESTORE=NOP,TAPE=RUN
/*
/&
```

NOTE: Once the tape is positioned, any appropriate restore function may be specified.

Emulate Tape

DR.D Emulate Tape Feature

Dr.D has the ability to use a disk file in place of tape. This feature can be useful in a number of situations, each of which will be discussed in detail:

- Unattended Backup
- Generic Processing
- VSAM-to-SAM or SAM-to-VSAM Conversion

Invoke the Tape Emulation feature by using DLBL/EXTENT JCL statements instead of TLBL. 'TAPEIN' and 'TAPEOUT' are the only 'DD' names allowed (the 'TDDNAME=' parameter is ineffective). If 'DTAPE' is used, only one file should be saved/restored in each Dr.D execution.

UNATTENDED BACKUP: The backup can take place after hours with no mounting or dismounting of tapes required, avoiding the necessity to stop online processing for backups. Additionally, VSAM reorganization can be accomplished without operator intervention. Later, the disk file can be copied to tape for archival storage while online processing continues.

GENERIC PROCESSING: Certain functions require the use of tape as intermediate storage, often preventing the use of those functions due to the unavailability of tape drives. For example, the 'SAVE/RESTORE=LOGICAL' function does not operate in 'COPY' mode, but is useful for creating test files with its 'RENAME' capability.

VSAM-TO-SAM OR SAM-TO-VSAM CONVERSION: If the 'TAPEIN' or 'TAPEOUT' file is disk and the 'DTAPE' (data tape) option is in effect, the file is created or read as a user format file (Fixed block or Variable block format). This allows files to be copied from one format to another. If 'DTAPE' is not specified, the file is created in Dr.D 'segmented' format and cannot be processed by user programs, and is expected to be in that format if being input to Dr.D.

For a CKD disk, use the 'DLBL BLOCK SIZE' parameter to specify an appropriate disk blocksize (use track length if possible). The disk block size is independent of the assumed or specified Dr.D 'BLOCKSIZE=' parameter (continue to use BLOCKSIZE as though tape was in use), and tape compression (CP=YES) will reduce the size of the disk file required to contain the backup. Standard IBM LIOCS is used, allowing disk manager control of the TAPEIN/TAPEOUT disk file.

Dr.D provides a disk-to-tape program (DOCTOR58) to copy the disk file to tape using 'DLBL TAPEIN' and 'TLBL TAPEOUT (SYS007)'. If 'UPSI 111' is set, DOCTOR58 will log information about each file/object copied to tape (see messages DOCTOR 250-252).

Alien-Backup-Tape Restore

Dr.D has the ability to restore backup tapes created by certain other disk utility systems, with certain restrictions. At this time, tapes can be restored that are produced by the Westinghouse disk utility (DUS, hereafter known as 'F2') and disk utility system product (DUSP, hereafter known as 'F1'), and the Computer Associates DYNUTIL (hereafter known as 'F3') and disk utility systems product (DUSP 2.0.1 / MAXBACK restore utility hereafter known as 'F4'), and the Legent FAVER VSAM utility (hereafter known as 'F5'). File types supported are VSAM, SAM, and Direct Access. Volume backups are supported for 'F1', and volume backups can be restored standalone without Dr.D involvement. VSAM files can be 'auto-defined' if produced by 'F1', 'F4' or 'F5', but must be user defined if produced by 'F2'. The 'Fn' format must be designated on a 'RESTORE=NOP' control card preceding the first 'RESTORE=VSAM/ DATA/SD' control card. Once the type is established, all other applicable Dr.D commands and keywords are effective, just as though the tape was produced by Dr.D (but RESTORE=LOGICAL is not supported- use individual restore control cards). The intent of this feature is to support all tape formats created by F1, F2, F4 or F5, but due to the chaotic nature of the development of the products, variations in tape format might require additional development within Dr.D.

Example 1. Restore from 'F1' format tape.

```
// JOB RESTORE ALIEN F1
// ASSGN SYS008,181
// MTC REW,SYS008
// TLBL TAPEIN
// DLBL X,'IM.IM.P03.IMBCRM.V12.BCR',,VSAM,CAT=IJSYSCT
// DLBL A,'F1.SD',0,SD CISIZE=30720
// EXTENT SYS001,,1,.,9435,1590
// DLBL B,'F1.SD2',0,SD
// EXTENT SYS001,,1,0,7012,500
// DLBL C,'F1.SD3',0,SD
// EXTENT SYS001,,1,0,7512,500
// ASSGN SYS001,472
// EXEC DRD,SIZE=200K
    RESTORE=NOP,UPSI=111,PRINT=YES,TAPE=F1
    RESTORE=VSAM,RFILE=X,AUTO=YES,FDELETE=DELETE
    RESTORE=SD,RFILE=A,LR=905,NREC=8
    RESTORE=DATA,RFILE=B
    RESTORE=DATA,RFILE=C
/@
/&
```

Example 2. Restore from 'F2' format tape.

```
// JOB RESTORE ALIEN F2
// MTC REW,181
// ASSGN SYS008,181
// TLBL TAPEIN
// DLBL X,'F2.VSAM',,VSAM,CAT=IJSYSUJ
// DLBL A,'F2.SD',1,SD,CISIZE=30720
// EXTENT SYS001,,1,0,12,16000
// ASSGN SYS001,26B
// EXEC DRD,SIZE=200K
   RESTORE=NOP,UPSI=111,PRINT=YES,BLOCKSIZE=65496,TAPE=F2
   RESTORE=VSAM,RFILE=X,FIND=SFFLE1,VRTYPE=ESDS
   RESTORE=DATA,RFILE=A,FIND=WISCRN,LB=2420
/@
/&
```

Example 3. Restore from 'F3' format tape.

```
// JOB RESTORE ALIEN F3
// MTC REW,181
// ASSGN SYS008,181
// TLBL TAPEIN
// DLBL A,'F3.SD',1,SD,CISIZE=30720
// EXTENT SYS001,,1,0,12,16000
// ASSGN SYS001,26B
// EXEC DRD,SIZE=200K
   RESTORE=NOP,UPSI=111,PRINT=YES,BLOCKSIZE=65496,TAPE=F3
   RESTORE=DATA,RFILE=A,FIND='F3.SD'
/@
/&
```

Example 4. Restore from 'F4' format tape.

```
// JOB RESTORE ALIEN F4
// ASSGN SYS008,181
// MTC REW,SYS008
// TLBL TAPEIN
// DLBL X,'IM.IM.P03.IMBCRM.V12.BCR',,VSAM,CAT=IJSYSCT
// DLBL A,'F4.SD',0,SD CISIZE=30720
// EXTENT SYS001,,1,.,9435,1590
// ASSGN SYS001,472
// EXEC DRD,SIZE=200K
    RESTORE=NOP,UPSI=111,PRINT=YES,TAPE=F4
    RESTORE=VSAM,RFILE=X,AUTO=YES,FDELETE=DELETE
    RESTORE=SD,RFILE=A,LR=905,NREC=8
/@
/&
```

Example 5. Restore from 'F1' volume backup.

```
// JOB RESTORE ALIEN F1 VOLUME BACKUP
// ASSGN SYS008,181
// MTC REW,SYS008
// TLBL TAPEIN
// EXEC DRD,SIZE=200K
   RESTORE=NOP,UPSI=111,PRINT=YES,TAPE =F1
   RESTORE=CURRENT,RV=DISK6T
   RESTORE=NO,TA=RU
/*
/&
```

Example 6. Restore from 'F5' format tape.

```
// JOB RESTORE ALIEN F5
// ASSGN SYS008,181
// MTC REW,SYS008
// TLBL TAPEIN
// DLBL X,'IM.IM.P03.IMBCRM.V12.BCR',,VSAM,CAT=IJSYSCT
// EXEC DRD,SIZE=200K
   RESTORE=NOP,UPSI=111,PRINT=YES,TAPE=F5
   RESTORE=VSAM,RFILE=X,FIND='AAA.DRD.TEST.FILE'
   RESTORE=NO,TA=RU
/*
/&
```

Dr.D Data Tape Feature

The Data Tape feature (keyword DTAPE=YES), somewhat similar to the IDCAMS program REPRO function, allows users to SAVE or RESTORE data files in user format with standard labels. The disadvantages to using this feature are loss of the VSAM automatic definition feature during the restore process, loss of the SAVE/RESTORE=LOGICAL function, and each file being saved stored in a unique save file. The file label being saved is identified with the SFILENAME keyword. The file label being restored is identified with the RFILENAME keyword. Specification of DTAPE=YES on a restore function is *only* allowed when the save file was created with DTAPE=YES. (See the *Sort Function for Save Files* to create a data file in user format from a save file not created with DTAPE=YES).

When using the Data Tape feature and saving to or restoring from disk, only one save/restore request per execution of Dr.D is allowed. When using the Data Tape feature and saving to or restoring from tape, multiple save/restore requests per execution of Dr.D are allowed since the tape is closed at the completion of each save/restore then re-opened at the start of the next file.

For the first tape file only, you may specify a unique filename using the TDDNAME keyword or allow Dr.D to use the default TAPEIN/TAPEOUT filename. Thereafter the filenames are generated in the format of 'TDInnnn' (input tape) or 'TDOnnnn' (output tape) allowing Dr.D to stack multiple files on the tape, if so desired. In either case the 'nnnn' represents a sequential number starting at 0001. This technique may be applied to any file type that is in fixed (blocked or unblocked) or variable (blocked or unblocked) format, VSAM, SAM, or any other file organization processed by Dr.D on a logical block or logical record basis.

Once saved, the data may be restored to any desired file type, allowing, for example, conversion of SAM files to VSAM files. When restoring, both the LR and NR keywords must be specified for fixed length records and the LB and VARIABLE=YES keywords must be specified for variable length records. When restoring to SAM files, reblocking may be accomplished by adjusting the value specified for the NR keyword (for fixed length records) or the LB keyword (for variable length records).

Using the SAVE file.

Although intended for use by Dr.D only, user written programs can read the SAVE file created with the Data Tape feature. The actual blocksize found in the SAVE file will approach the value specified (or defaulted) for the BLKSIZE keyword, but never actually reach that value. Dr.D always builds internal buffers with all potential control information, which varies with the file organization and file format as well the device type on which the file resides. All that control information, however, is removed before the buffer is actually written to the save file. If you choose to read the SAVE file for fixed length records, choose a blocksize value equal to the largest multiple of the logical record size that does not exceed the value specified (or defaulted) for the BLKSIZE keyword. If you

choose to read the SAVE file for variable length records, choose a blocksize value equal to the value specified (or defaulted) for the BLKSIZE keyword. Use of these techniques will always satisfy Logical IOCS requirements.

Example 1. Save four VSAM files to tape with Data Tape feature.

This example saves four VSAM files using the Data Tape feature. It assumes the files are all defined with fixed length records.

```
// JOB P01 SAVE VSAM USING THE DATA TAPE FEATURE
// ASSGN SYS007,181
// MTC REW,181
// DLBL FIL1,'AR.CUST.NA',,VSAM,CAT=IJSYSUJ
// DLBL FIL2,'ASSISAM',,VSAM,CAT=IJSYSUJ
// DLBL FIL3,'CAMAST.FOUR.SIX',,VSAM,CAT=IJSYSUJ
// DLBL FIL4,'LABS.SAMPLES.FILE',,VSAM,CAT=IJSYSUJ
// TLBL TAPEOUT,'AR.CUST.NA'
// TLBL TDO0001,'ASSISAM'
// TLBL TDO0002,'CAMAST.FOUR.SIX'
// TLBL TDO0003,'LABS.SAMPLES.FILE'
// EXEC DRD,SIZE=200K
SAVE=VSAM,SFILE=FIL1,DTAPE=YES,FAST=YES,TAPE=REW,UPSI=111
SAVE=VSAM,SFILE=FIL2,DTAPE=YES,FAST=YES
SAVE=VSAM,SFILE=FIL3,DTAPE=YES,FAST=YES
SAVE=VSAM,SFILE=FIL4,DTAPE=YES,FAST=YES
SAVE=NOP,TAPE=REW
/*
/&
```

Example 2. Restore four VSAM files with Data Tape feature.

This example restores four VSAM files saved in example 1. It assumes all the VSAM files are either null or have been defined as reusable.

```
// JOB P02 RESTORE VSAM USING THE DATA TAPE FEATURE
// ASSGN SYS008,181
// MTC REW,181
// TLBL TAPEIN,'AR.CUST.NA'
// TLBL TDI0001,'ASSISAM'
// TLBL TDI0002,'CAMAST.FOUR.SIX'
// TLBL TDI0003,'LABS.SAMPLES.FILE'
// DLBL FIL1,'AR.CUST.NA',,VSAM,CAT=IJSYSUJ
// DLBL FIL2,'ASSISAM',,VSAM,CAT=IJSYSUJ
// DLBL FIL3,'CAMAST.FOUR.SIX',,VSAM,CAT=IJSYSUJ
// DLBL FIL4,'LABS.SAMPLES.FILE',,VSAM,CAT=IJSYSUJ
// EXEC DRD,SIZE=200K
RESTORE=VSAM,RFILE=FIL1,DTAPE=YES,TAPE=REW,UPSI=111
RESTORE=VSAM,RFILE=FIL2,DTAPE=YES
RESTORE=VSAM,RFILE=FIL3,DTAPE=YES
RESTORE=VSAM,RFILE=FIL4,DTAPE=YES
RESTORE=NOP,TAPE=RUN
/*
/&
```

Example 3. Save two SD Files with Data Tape feature.

This example saves two SD files using the Data Tape feature.

```
// JOB SAVE AND RESTORE SD FILES USING THE DATA TAPE FEATURE
// ASSGN SYS007,181
// MTC REW,181
// DLBL FIL1,'FIXED.BLOCKED.SAM.FILE',,SD
// EXTENT SYS001,,1,0,15,300
// DLBL FIL2,'VARIABLE.BLOCKED.SAM.FILE',,SD
// EXTENT SYS001,,1,0,4500,6000
// TLBL TAPEOUT,'FIXED.BLOCKED.SAM.FILE'
// TLBL TDO0001,'VARIABLE.BLOCKED.SAM.FILE'
// EXEC DRD,SIZE=200K
SAVE=DATA,SFILE=FIL1,DTAPE=YES,TAPE=REW,UPSI=111
SAVE=DATA,SFILE=FIL2,DTAPE=YES
SAVE=NOP,TAPE=REW
/*
/&
```

Example 4. Restore SD files saved with the Data Tape feature.

This example restores the two SD files saved in example 3 reblocking each file.

```
// JOB RESTORE SD FILES USING THE DATA TAPE FEATURE
// ASSGN SYS008,181
// MTC REW,181
// DLBL FIL1,'FIXED.BLOCKED.SAM.FILE',,SD
// EXTENT SYS001,,1,0,15,300
// DLBL FIL2,'VARIABLE.BLOCKED.SAM.FILE',,SD
// EXTENT SYS001,,1,0,4500,6000
// TLBL TAPEIN,'FIXED.BLOCKED.SAM.FILE'
// TLBL TDI0001,'VARIABLE.BLOCKED.SAM.FILE'
// EXEC DRD,SIZE=200K
RE=DATA,RFILE=FIL1,DTAPE=YES,TA=REW,UPSI=111,LREC=100,NRECDS=100
RESTORE=DATA,RFILE=FIL2,DTAPE=YES,VARIABLE=YES,LBLOCK=16000
RESTORE=NOP,TAPE=REW
/*
/&
```

GETTAPE Sub-Program

The GETTAPE subroutine can be called from a user program to access a data file found on a Dr.D save file. The save file must be on tape – this subroutine will fail trying to open a save file stored on disk. This user program into which this subroutine is included can be relocatable, self-relocating, or linked at a fixed address as long as standard calling protocol is observed, such as produced by COBOL and ASSEMBLER. Use of this subroutine eliminates any need to understand the complexities of the Dr.D save tape, including identification of embedded control information and compression techniques.

There are 4 functions included in the GETTAPE subroutine:

- SETTAPE
- PASSCARD
- GETTAPE
- CLOSTAPE

SETTAPE Function

The SETTAPE function provides for specifying the programmer logical unit number to be used to read the Dr.D save tape, for specifying the filename to be used to identify the save tape, and to identify the data file residing on the save tape. If used, the SETTAPE function must be called prior to the first call to the GETTAPE function and all fields must be supplied. If no call is made to the SETTAPE function, this subroutine will default to programmer logical unit SYS008 for filename TAPEIN and will read the first data file found on the save tape. The data area passed when calling the SETTAPE function is as follows:

Start Position	Length	Field Description
1	6	Programmer logical unit number in the form SYSnnn where nnn is a numeric value from 000 through 255.
7	7	Filename of the TLBL used to identify the save tape, left justified and padded with blanks if less than 7 characters.
14	46	Filename or file-id of data file residing on the save tape. If specifying a filename, use only the first 7 positions of this field and pad with blanks if less than 7 characters. If specifying a file-id, place a single quote in the first position, the 44-character file-id in the next 44 positions (left justified and padded with blanks if less than 44 characters), and a single quote in the last position.

Assembler Example of call to SETTAPE with FIND for File-ID:

```
         CALL  SETTAPE,(NEWOPTS)
         .
         .
         .
NEWOPTS  DS    0CL59
         DC    CL6'SYS010'
         DC    CL7'SVTAPE '
         DC    C''''
         DC    CL44'MY DATASET FILE-ID'
         DC    C''''
```

Assembler Example of call to SETTAPE with FIND for Filename:

```
        CALL  SETTAPE,(NEWOPTS)
         .
         .
         .
NEWOPTS  DS    0CL20
         DC    CL6'SYS010'
         DC    CL7'SVTAPE '
         DC    CL7'DISKFLE'
```

COBOL Example of call to SETTAPE with FIND for File-ID:

```
        01  NEW-OPTIONS.
            03  FILLER      PIC X(6)        VALUE 'SYS010'.
            03  FILLER      PIC X(7)        VALUE 'SVTAPE'.
            03  FILLER      PIC X           VALUE QUOTE.
            03  FILLER      PIC X(44)       VALUE 'MY.DATASET.FILE-ID'.
            03  FILLER      PIC X           VALUE QUOTE.
            .
            .
            .
            CALL 'SETTAPE' USING NEW-OPTIONS.
```

COBOL Example of call to SETTAPE with FIND for Filename:

```
        01  NEW-OPTIONS.
            03  FILLER      PIC X(6)        VALUE 'SYS010'.
            03  FILLER      PIC X(7)        VALUE 'SVTAPE'.
            03  FILLER      PIC X(7)        VALUE 'DISKFLE'.
            .
            .
            .
            CALL 'SETTAPE' USING NEW-OPTIONS.
```

PASSCARD Function

The PASSCARD function provides for identifying the format of the data file to be read from the save tape. If used, the PASSCARD function must be called prior to the first call to the GETTAPE function. If no call is made to the PASSCARD function prior to the first call to the GETTAPE function, a control card will be read from SYSIPT (using a standard DTFDI) to obtain this information, using the exact same format as described below. The data area passed when calling the PASSCARD function is as follows:

Start Position	Length	Field Description		
1	4	TAPE		
5	2	**Identifier**	**File Format**	
		VS	VSAM	
		IS	ISAM	
		FB	Fixed Length Blocked	
		FU	Fixed Length Unblocked	
		VB	Variable Length Blocked	
		VU	Variable Length Unblocked	
		UN	Undefined	
7	8	**Identifier**	**Value(s)**	
		VS	Not used	
		IS	Not used	
		FB	bbbbb	5-digit unsigned zoned decimal blocksize
			lll	3-digit unsigned zoned decimal blocking factor (i.e., number of logical records per block)
		FU	bbbbb	5-digit unsigned zoned decimal blocksize
			kkk	3-digit unsigned zoned decimal key length (Use 000 if no key)
		VB	bbbbb	5-digit unsigned zoned decimal blocksize
			xxx	Not used
		VU	bbbbb	5-digit unsigned zoned decimal blocksize
			kkk	3-digit unsigned zoned decimal key length (Use 000 if no key)
		UN	Not used	

Assembler Example of call to PASSCARD for Fixed Blocked Records:

```
         CALL  PASSCARD,(FILEINFO)
         .
         .
         .
FILEINFO DS    0CL14
         DC    CL4'TAPE'
         DC    CL2'FB '
         DC    CL5'00800'
         DC    CL3'010'
```

Assembler Example of call to PASSCARD for VSAM records:

```
         CALL  PASSCARD,(FILEINFO)
         .
         .
         .
FILEINFO DS    0CL6
         DC    CL4'TAPE'
         DC    CL2'VS '
```

COBOL Example of call to PASSCARD for Fixed Blocked Records:

```
       01  FILE-INFO.
           03  FILLER      PIC X(4)        VALUE 'TAPE'.
           03  FILLER      PIC X(2)        VALUE 'FB'.
           03  FILLER      PIC X(5)        VALUE '00800'.
           03  FILLER      PIC X(3)        VALUE '010'.
           .
           .
           .
           CALL 'PASSCARD' USING FILE-INFO.
```

COBOL Example of call to PASSCARD for VSAM Records:

```
       01  FILE-INFO.
           03  FILLER      PIC X(4)        VALUE 'TAPE'.
           03  FILLER      PIC X(2)        VALUE 'VS'.
           .
           .
           .
           CALL 'PASSCARD' USING FILE-INFO.
```

GETTAPE Function

The GETTAPE function yields the next logical record from the data file identified (or defaulted to) in a preceding call to the SETTAPE function and described either by a preceding call to the PASSCARD function or by a control card provided as SYSIPT input. The first call to the GETTAPE function will accomplish opening of the save tape without any repositioning. Therefore, the save tape must have already been properly positioned. Data from each call to the GETTAPE function is stored in your program's receiving area in the following formats:

VSAM KSDS and ESDS with Fixed Length Records saved with SAVE=VSAM and VARIABLE=NO		
Start	**Length**	**Field Description**
1	*x*	Actual Record
VSAM KSDS and ESDS saved with SAVE=LOGICAL or VSAM KSDS and ESDS saved with SAVE=VSAM and VARIABLE=YES		
Start	**Length**	**Field Description**
1	2	Actual Record Length + 4 (Binary)
3	2	Binary Zeros
5	*x*	Actual Record
VSAM RRDS with Fixed Length Records saved with SAVE=VSAM and VARIABLE=NO		
Start	**Length**	**Field Description**
1	2	Relative Record Number (Binary)
3	*x*	Actual Record
VSAM RRDS saved with SAVE=LOGICAL or VSAM RRDS saved with SAVE=VSAM and VARIABLE=YES		
Start	**Length**	**Field Description**
1	2	Actual Record Length + 8 (Binary)
3	2	Binary Zeros
5	4	Relative Record Number (Binary)
9	*x*	Actual Record
Blocked Fixed Length Records		
Start	**Length**	**Field Description**
1	*x*	Actual Record

VSAM KSDS and ESDS with Fixed Length Records saved with SAVE=VSAM and VARIABLE=NO		
Start	**Length**	**Field Description**
Unblocked Fixed Length Records		
Start	**Length**	**Field Description**
1	*k*	Key (If any)
1+*k*	*x*	Actual Record
Blocked and Unblocked Variable Length Records		
Start	**Length**	**Field Description**
1	2	Key Length (Zero if no key)
3	2	Binary Zeros
5	2	Actual Record Length + 4 (Binary)
7	2	Binary Zeros
9	*k*	Key (If any)
9+*k*	*x*	Actual Record
Undefined Length Records		
Start	**Length**	**Field Description**
1	2	Key Length (Zero if no key)
3	2	Binary Zeros
5	2	Actual Record Length (Binary)
7	2	Binary Zeros
9	*k*	Key (If any)
9+*k*	*x*	Actual Record

Regardless of file organization or record format, end of data file is always indicated by the 16-character literal '`END OF DATA FILE`' placed in positions 1 through 16 of your program's receiving area. Once end of data file has been indicated, only a single call to the CLOSTAPE function is allowed.

Assembler Example of call to GETTAPE:

```
         CALL  GETTAPE,(DATAREA)
         CLC   =CL16'END OF DATA FILE',DATAREA
         BE    DFILEOF
         .
         .
         .
DATAREA  DS    CL..
```

COBOL Example of call to GETTAPE:

```
       01  DATA-AREA.
           .
           .
           .
       01  FILLER REDEFINES DATA-AREA.
           03  FILLER      PIC X(16).
               88  DA-END-OF-DATA-FILE     VALUE 'END OF DATA FILE'.
           03  FILLER      PIC X(…).
           .
           .
           .
           CALL 'GETTAPE' USING DATA-AREA.
           IF DA-END-OF-DATA-FILE
               GO TO END-OF-DATA-FILE.
```

CLOSTAPE Function

The CLOSTAPE function closes the save tape and should be the final call made to this sub-program. The tape is not repositioned when the close is accomplished. No data area is passed when call the CLOSTAPE function.

Assembler Example of call to CLOSTAPE:

```
         CALL  CLOSTAPE
```

COBOL Example of call to CLOSTAPE:

```
           CALL  'CLOSTAPE'.
```

GETTAPE Example 1.

Using COBOL, read a file to display the total of the values found in a certain field. The file has been saved by Dr.D and contains fixed 100-byte logical length records blocked 4 giving 400-byte physical records.

COBOL Source:

```
       ID DIVISION.
       PROGRAM-ID. 'SUMTAPE'.
       ENVIRONMENT DIVISION.
       DATA DIVISION.
       WORKING-STORAGE SECTION.
       01 CHANGE-THE-SYSNUM-AND-FILENAME.
          02 SYSNUM     PIC X(6) VALUE 'SYS003'.
          02 FILENAME   PIC X(7) VALUE 'MASTER'.
          02 FINDNAME   PIC X(7) VALUE 'DDNAME '.
       01 THE-SUM       PIC 9(7) COMP-3 VALUE ZERO.
       01 THE-RECORD-FROM-GETTAPE.
          02 A-CERTAIN-FIELD              PIC 9(5).
          02 OTHER-STUFF                  PIC X(95).
       01 FILLER REDEFINES THE-RECORD-FROM-GETTAPE.
          02 FILLER                         PIC X(16).
          88  EOD VALUE 'END OF DATA FILE'.
       PROCEDURE DIVISION.
           CALL 'SETTAPE' USING
                 CHANGE-THE-SYSNUM-AND-FILENAME.
       GETTAPE-LOOP.
           CALL 'GETTAPE' USING
               THE-RECORD-FROM-GETTAPE.
           IF NOT EOD
               ADD A-CERTAIN-FIELD TO THE-SUM
               GO TO GETTAPE-LOOP.
           DISPLAY THE-SUM UPON CONSOLE.
           CALL 'CLOSTAPE'.
           STOP RUN.
```

Execution JCL:

```
// JOB SUMTAPE
// TLBL MASTER
// ASSGN SYS003,280
// EXEC SUMTAPE
TAPEFB00400004
/*
/&
```

GETTAPE Example 2.

Using COBOL, read a file to display the total number of salesmen that have sold more widgets than a figure supplied on a SYSIPT input transaction. This KSDS file contains fixed 90-byte length records and has been saved by Dr.D using SAVE=VS and VARIABLE=NO.

COBOL Source:

```
        IDENTIFICATION DIVISION.
        PROGRAM-ID.  'CNTIT'.
        ENVIRONMENT DIVISION.
        DATA DIVISION.
        WORKING-STORAGE SECTION.
        77    NUMBER-OF-SALESMEN    PIC X(6) COMP-3 VALUE 0.
        01    THE-CARD.
              02   WIDGETS          PIC X(5)
              02   FILLER           PIC X(75).
        01    THE-MASTER-RECORD.
              02   KEY-IS-SALESMAN  PIC X(5).
              02   SALESMAN-NAME    PIC X(30).
              02   SALESMAN-ADR     PIC X(50).
              02   WIDGETS-SOLD     PIC X(5).
              02   A-BUNCH-OF-OTHER-FIELDS  PIC X(100).
        01    FILLER REDEFINES THE-MASTER-RECORD.
              02 FILLER             PIC X(16).
                 88  EOD VALUE 'THE END OF DATA FILE'.
        PROCEDURE DIVISION.
              ACCEPT THE-CARD.
        CALL-GETTAPE-LOOP.
              CALL 'GETTAPE' USING THE-MASTER-RECORD.
              IF EOD GO TO THE-END-OF-FILE.
              IF WIDGETS-SOLD GREATER THAN WIDGETS
                 ADD 1 TO NUMBER-OF-SALESMEN.
              GO TO CALL-GETTAPE-LOOP.
        THE-END-OF-FILE.
              DISPLAY 'SALESMEN WHO SOLD MORE THAN'
                 WIDGETS 'WIDGETS IS' NUMBER-OF-SALESMEN
                 UPON CONSOLE.
              STOP RUN.
```

Execution JCL:

```
// JOB CNTIT
// TLBL TAPEIN
// ASSGN SYS008,280
// EXEC CNTIT
07500
TAPEVS
/*
/&
```

Sort Function for Backup Files

The Dr.D sort function allows the data portion of a saved file on a tape or disk backup file (created by Dr.D) to be extracted and passed as input to the user's SORT program. Multiple sort functions may be requested within a single Dr.D execution but only one data file is extracted per request. This function operates much like a Dr.D restore function, and restore functions may be intermixed with the sort function as well as other functions.

The RFILENAME keyword is used to designate the file to be specified as 'SORTOUT' to the user's SORT program. Use the FIND keyword to position to the desired saved file. The files must be processed in the order that they were saved, and the use of UPSI XXXXXX1 is not allowed.

The SORT program control statements must immediately follow the Dr.D sort function request where each occupies up to the leading 79 positions of each 80-byte length record with no control statement continuation and properly terminated by either a '/*' or '/@'. The output file format reflected on the TYPE= parameter value required on the SORT program RECORD control statement must match the format of the saved data or the results will be unpredictable. Only the following SORT program control statements are allowed (others are ignored):

- ALTSEQ
- ANALYZE
- INCLUDE
- INREC
- OMIT
- OPTION
- OUTFIL
- OUTREC
- RECORD
- SORT
  - A copy will result with FIELDS=COPY
  - A sort will result with FIELDS=(p,m,…)
- SUM

Partition and GETVIS requirements will vary depending on the user's SORT program and whether SVA eligible phases have been loaded into the SVA.

Example 1. Create five data files from backup tape.

This example creates five data files from a Dr.D backup tape.

```
// JOB SORT FILES ON BACKUP TAPE
// TLBL TAPEIN..............
// DLBL SORTWK1,'SORT.WORK.1'
// EXTENT ............................
// DLBL SORTOUT,'SORT.OUTPUT.FILE.1'
// EXTENT ............................
// DLBL   SORT2,'SORT.OUTPUT.FILE.2'
// EXTENT ............................
// DLBL   SORT3,'SORT.OUTPUT.FILE.3'
// EXTENT ............................
// DLBL   SORT4,'SORT.OUTPUT.FILE.4'
// EXTENT ............................
// DLBL   SORT5,'SORT.OUTPUT.FILE.5'
// EXTENT ............................
// EXEC DRD,SIZE=320K
   FUNCTION=SORT,RFILENAME=SORTOUT,FIND=FIL0001
     SORT FIELDS=(4,10,A),FORMAT=BI,WORK=1
     RECORD TYPE=V,LENGTH=(200,,,100,170)
     OUTFIL BLKSIZE=8192
     OPTION PRINT=ALL
   /@
   FUNCTION=SORT,RFILENAME=SORT2,FIND='PARTS.MASTER'
     SORT FIELDS=(4,10,A),FORMAT=BI,WORK=1
     RECORD TYPE=V,LENGTH=(200,,,100,170)
     OUTFIL BLKSIZE=8192
     OPTION PRINT=ALL
   /@
   FUNCTION=SORT,RFILENAME=SORT3,FIND=FIL0006
     SORT FIELDS=(4,10,A),FORMAT=BI,WORK=1
     RECORD TYPE=F,LENGTH=40
     OUTFIL BLKSIZE=8120
     OPTION PRINT=ALL
   /@
   FUNCTION=SORT,RFILENAME=SORT4,FIND=FIL0100
     SORT FIELDS=COPY
     RECORD TYPE=F,LENGTH=196
     OUTFIL BLKSIZE=8232
     OPTION PRINT=ALL
   /@
   FUNCTION=SORT,RFILENAME=SORT5,FIND=FIL0101
     SORT FIELDS=(4,10,A),FORMAT=BI,WORK=1
     RECORD TYPE=V,LENGTH=(200,,,100,170)
     OUTFIL BLKSIZE=8192
     OPTION PRINT=ALL
   /@
/*
/&
```

Dr.D Listtape Function

Example 1. List contents of suspect tape.

This job stream reports on the contents of suspect tapes in terms of files, volumes, libraries, etc.

```
//  JOB SCAN A SAVE TAPE
//  ASSGN SYS008,...                    TAPE
//  TLBL TAPEIN
//  UPSI 0           ALLOW SYSLOG OUTPUT
//  UPSI 1           PREVENT SYSLOG OUTPUT
//  EXEC DRD,SIZE=200K
RESTORE=NOP,BLOCKSIZE=65496,PRINT=YES
FUNCTION=LTAPE
/*
/&
```

All of the following is noted on the printer and / or SYSLOG:

(1) presence of standalone IPL records on the front of the tape,
(2) the time and date the tape was created,
(3) a description of each save that was done to create the tape.

If '// UPSI 1' is used, the report is produced on SYSLST only. If execution is abnormally terminated via an illegal SVC 255, there is a format error on the save tape. This probably means the tape has been scratched or it's not a save tape.

Dr.D Tape Copy Feature

Dr.D can copy tapes in two different modes. The first copies at the logical tape file level, with an open and close for each input and output tape. The second copies in 'image' mode- does not open or close tapes, and expects each input tape to fit within the limits of its corresponding output tape. In either mode, tape blocks can be modified by supplying pairs of scan and replace control cards.

Dr.D Logical Tape Copy Feature

The logical tape copy function allows single or multi-volume tapes to be copied, and if desired, reblocked in the process. Normal Dr.D tape commands are used to REWIND/UNLOAD tapes, and stacked (multi- file) tapes may be processed since tapes are normally opened and closed with NO REWIND. Input and output tapes are assumed to have standard labels, but input may be unlabeled if the 'NO VOL1' message is 'ignored'.

Example 1. Tape Copy without Reblock.

```
// JOB TAPE COPY WITHOUT REBLOCK
// ASSGN SYS007,181   OUTPUT TAPE DRIVE
// ASSGN SYS008,180   INPUT TAPE DRIVE
// TLBL TAPEIN
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
   FUNCTION=CTAPE
   RESTORE=NONE,TAPE=RUN      UNLOAD INPUT TAPE
   SAVE=NONE,TAPE=RUN         UNLOAD OUTPUT TAPE
/*
/&
```

Example 2. Tape Copy- Fixed Blocked with Reblock.

```
// JOB TAPE COPY FIXED BLOCKED WITH REBLOCK
// ASSGN SYS007,181   OUTPUT TAPE DRIVE
// ASSGN SYS008,180   INPUT TAPE DRIVE
// TLBL TAPEIN
// TLBL TAPEOUT
// EXEC DRD,SIZE=200K
   FUNCTION=CTAPE,LRECORDSIZE=80,NRECORDS=100
   RESTORE=NONE,TAPE=RUN      UNLOAD INPUT TAPE
   SAVE=NONE,TAPE=RUN         UNLOAD OUTPUT TAPE
/*
/&
```

Example 3. Tape Copy- Variable Blocked with Reblock creating two copies.

```
// JOB TAPE COPY VARIABLE BLOCKED WITH REBLOCK
// ASSGN SYS007,181   OUTPUT TAPE DRIVE
// ASSGN SYS006,182   2ND OUTPUT TAPE DRIVE
// ASSGN SYS008,180   INPUT TAPE DRIVE
// TLBL TAPEIN
// TLBL TAPEOUT
// TLBL TAPETWO
// EXEC DRD,SIZE=200K
   FUNCTION=CTAPE,LBLOCKSIZE=8000,UP=0001
   RE=NONE,TAPE=RUN      UNLOAD INPUT TAPE
   SA=NONE,TAPE=RUN      UNLOAD OUTPUT TAPES
/*
/&
```

Dr.D Image Mode Tape Copy Feature

The Dr.D 'image' mode tape copy makes an exact copy of input tapes including any 'VOL' 'HDR' or trailer labels. Since the tapes are not opened, a warning message (DOCTOR 610) is issued before the first copy begins so that the operator will know to be especially careful to insure that only scratchable tapes are mounted on the output drive(s). The user should be aware that the VOL label is copied exactly so that there is then more than one tape with the same volume serial number, and be aware of possible problems if a tape management system is in use. This copy method copies data and tapemarks until: 1) the EOT reflector is sensed on input or output or 2) a tape error occurs on the input tape. If end of output occurs, the operator will be asked whether to treat as end of input or to continue on another tape. If end of input (or tape error) occurs, it is treated as end of copy. For information on how to use the 'FORMAT=' parameter please refer to the keyword section of this manual.

Example 4. Image Tape Copy.

```
// JOB IMAGE TAPE COPY
// ASSGN SYS007,181   OUTPUT TAPE DRIVE
// ASSGN SYS008,180   INPUT TAPE DRIVE
// EXEC DRD,SIZE=200K
   FUNCTION=CTAPE,FORMAT=IMAGE,TAPES=2
/*
/&
```

Example 5. Image Tape Copy- Dual output.

```
// JOB IMAGE TAPE COPY WITH TWO OUTPUT
// ASSGN SYS006,182   SECOND OUTPUT TAPE DRIVE
// ASSGN SYS007,181   OUTPUT TAPE DRIVE
// ASSGN SYS008,180   INPUT TAPE DRIVE
// UPSI 1111
// EXEC DRD,SIZE=200K
   FUNCTION=CTAPE,FORMAT=IMAGE,TAPES=1
/*
/&
```

Example 6. Image or logical tape copy with modification. MODIFY=YES is specified along with pairs of search and replace control cards.

```
            // JOB COPY TAPE WITH MODIFY
            // ASSGN SYS006,182   SECOND OUTPUT TAPE DRIVE
            // ASSGN SYS007,181   OUTPUT TAPE DRIVE
            // ASSGN SYS008,180   INPUT TAPE DRIVE
            // UPSI 1111
            // EXEC DRD,SIZE=200K
               FUNCTION=CTAPE,MODIFY=YES
search-->     C'OLD DATA'
replace->     C'NEW DATA'
search-->     X'05060708'
replace->     X'AABCDEFF'
               /@
            /*
            /&
```

Dr.D Tape Test Feature

This feature allows tapes to be tested by writing the chosen blocksize continuously over the length of the reel or cartridge. A report is produced showing the number of 'erase gaps' and 'unrecoverable errors' encountered for each cartridge or reel. If a VOL1 is found on the tape, it is printed on the report and retained on the tape. If no VOL1 is found, one is created using the sequence number listed on the report as the VOLUME SERIAL number of the tape. If the starting VOLSER is entered using 'VOLSER=nnnnnn', then each tape volume serial number is generated starting at the entered number, and is incremented by one for each tape processed.

For each tape tested, a map is printed showing how the errors are distributed over the tape. Each position of the map is a percentage point for showing the relative position and the number of errors occurring at that point. Thus if 5 errors occurred at the mid-point of the tape, a '5' would appear under the '50' heading. If more than 9 errors occur at any point, then multiple lines are printed, for example:

	8	indicates that 819 errors
occurred at that point.	1	
	9	

Either one or two tape drives can be used, so that tape rewind can be overlapped with tape testing. Since the tapes are not OPEN'd, a warning is issued stating the drives to be used, allowing the operator to cancel if a conflict exists.

The number of tapes to be tested can be controlled by specifying 'TAPES=NNN' on the control card, or by operator communication with Dr.D. The operator may also force early 'end of tape' by entering 'MSG PI' (PI is the partition ID, I.E. BG, F1, etc.) and then entering 'EOV'. To terminate tape testing at the end of current tape, enter 'ENDTEST' after entering the 'MSG' command.

Example 1. Tape Test using only one tape drive.

```
// JOB TAPE TEST USING ONE TAPE DRIVE
// ASSGN SYS007,180   ASSIGN DRIVE TO USE
// EXEC DRD,SIZE=200K
   FUNCTION=TAPETEST,TAPES=1
/*
/&
```

Example 2. Tape Test using two tape drives.

```
// JOB TAPE TEST USING TWO TAPE DRIVES
// ASSGN SYS007,180   ASSIGN 1ST DRIVE
// ASSGN SYS006,181   ASSIGN 2ND DRIVE
// UPSI 1111
// EXEC DRD,SIZE=200K
   FUNCTION=TAPETEST,TAPES=10
/*
/&
```

Stabilized Components

Saving, Restoring, Copying ISAM Data Files

To work with an ISAM file, specify 'SAVE/COPY/RESTORE=DATA/ISAM'.

Specify 'SFILENAME=' when 'SAVE/COPY=DATA' is specified to inform Dr.D of the files to be read. Likewise, specify 'RFILENAME' when 'COPY/RESTORE=DATA' is specified.

DLBL and extents must be provided for either in the standard labels or JCL for those files. If 'COPY=DATA', both 'SF=' and 'RF=' must be specified and they must be unique. 'FA=YES' invokes the special feature for fast ISAM writes which requires about 20K of additional memory.

The restore file's extents are completely independent of the save file's. They may differ in size, placement, split cylinder specifications, and types of disk drives. Additionally, ISAM files may be rebuilt with or without a master cylinder index - just by the extent card's presence or absence. By the use of 'NRECDS=XX', the blocking factor of an ISAM file may be altered upon restoration. Further, using 'CYLOFLO=XX' as the operand, will alter the file's cylinder overflow option upon restoring. Records may be deleted from an ISAM file by specifying 'DELETE=PPPPHH' where PPPP is the position (relative to one) of a one byte delete code and HH represents the code in hexadecimal. To delete all records having the character 'D' in position 15 of the record, use 'DE=0015C4'. For UNBLOCKED files, the relative location within the record must include the key area. If 'SAVE=ISAM' is used, logical IOCS is used to save the file. This allows PSAM files to be saved/copied by Dr.D.

A DTF is initialized as follows:

1. If SYS001 is assigned to disk, its device type is used for the prime data device.
2. If SYS000 is assigned to disk, its' device type is used for the master index device type when using a master index.
3. If SYS000/SYS001 is not assigned to disk, the SYSRES device type is used in either case.

Data tapes may be saved and restored for ISAM files instead of Dr.D format tapes. When saving, specify 'DT=YES' and the Data Tape will be produced in fixed block format. The blocksize will be the largest possible for the assumed or specified blocksize 'BL=NNNNN'. To change the blocking factor on the tape, specify 'BL=NNNNN' where 'NNNNN' is less than 32752 and is the blocking factor times the logical record length plus 10. The extra ten spaces are needed for boundary considerations and will not be used. Only as many complete logical records as can be placed in the size 'NNNNN' will be written to the tape. When restoring a Data Tape to an ISAM file, considerably more information is required than if a Dr.D save file is used. 'NR=' and 'CY=' parameters may not be used when restoring ISAM files from data tapes. The required parameter is

'DT=AABBBCCCDDDDDEEEEE' where 'AA' is the number of tracks for Cylinder Overflow, 'BBB' is the number of logical records per Physical Block, 'CCC' is the length of the key, 'DDDDD' is the length of the Logical Record, and 'EEEEE' is the position in the logical record where the key begins (counting from one). With either Dr.D or Data Tape input to a restore, FAST ISAM write may be invoked with 'FA=YES'.

For ISAM restores, the DLBL file type must be 'ISC', for SAVES it may be 'ISC' or 'ISE'(ISE only, if SAVE=ISAM).

ISAM files are reorganized when restored. If duplicate or out-of- sequence records are encountered, you have the option of 'IGNORE' or 'CANCEL'.

During save operations, out of sequence records are counted and the count displayed on SYSLOG if / / UPSI XX1 is set.

Example 6. Save ISAM file to tape.

This example saves an ISAM file to tape.

```
// JOB D6
// DLBL ISAMFLE,'MASTER',99/365,ISC
// EXTENT SYS001,111111,4,1,40,20
// EXTENT SYS002,222222,1,2,20,3980
// TLBL TAPEOUT
// ASSGN SYS007,280
// EXEC DRD,SIZE=200K
   SAVE=DATA,SFILE=ISAMFLE
/*
/&
```

Example 7. Restore ISAM file.

This example restores the file saved in example 6.

```
// JOB D7
// DLBL ISAMFLE,'MASTER',99/365,ISC
// EXTENT SYS001,111111,4,1,40,20
// EXTENT SYS002,222222,1,2,20,3980
// TLBL TAPEIN
// ASSGN SYS008,280
// EXEC DRD,SIZE=200K
   RESTORE=DATA,RFILE=ISAMFLE
/*
/&
```

Example 8. Copy Multivolume ISAM File from a 3350 to a 3375.

This example copies a multivolume ISAM file from a 3350 to a single 3375. It deletes all records having a HEX 'FF' in position 12 and changes the blocking from unblocked to three records per block. Assume the 3350 file has no cylinder overflow but has an independent overflow area. The job stream makes the cylinder overflow two tracks and removes the independent overflow area. The 3350 file has no master cylinder index. The job stream creates one for the 3375 file.

```
// JOB D8
// DLBL OLDFILE,'3350 ISAM',99/360,ISE
// EXTENT SYS001,000001,4,1,40,20
// EXTENT SYS002,000002,1,2,20,3980
// EXTENT SYS003,000003,1,3,20,3980
// EXTENT SYS001,000001,2,4,60,1000
// DLBL NEWFILE,'3375 ISAM',99/360,ISC
// EXTENT SYS004,888888,4,0,12,12
// EXTENT SYS004,888888,4,1,24,24
// EXTENT SYS004,888888,1,2,48,7200
// ASSGN SYS001,131
// ASSGN SYS002,132
// ASSGN SYS003,133
// ASSGN SYS004,1C1
// EXEC DRD,SIZE=200K
   CO=DA,SF=OLDFILE,RF=NEWFILE,DELETE=0012FF,CYLOFLO=02,NRECDS=03
/*
/&
```

Example 9. Save Multivolume ISAM File

Using logical IOCS, this job stream saves the ISAM file created in example 8. SYSRES is on a 3350. Employ SYS000 and SYS001 to set proper device types.

```
// JOB D9
// DLBL A,'3375 ISAM',99/360,ISE
// EXTENT SYS004,888888,4,0,12,12
// EXTENT SYS004,888888,4,1,24,24
// EXTENT SYS004,888888,1,2,48,7200
// ASSGN SYS000,1C1 insure MASTER INDEX is valid device
// ASSGN SYS001,1C1 insure PRIME DATA is valid device
// ASSGN SYS004,1C1
// TLBL TAPEOUT
// ASSGN SYS007,280
// EXEC DRD,SIZE=200K
   SAVE=ISAM,SFILE=A
/*
/&
```

Summary of Control Card Format for Data Files (ISAM)

Save Data File (ISAM):

```
SAVE=DATA,SFILE=FFFFFFF
(IS)
```

Restore Data File (ISAM):

```
RESTORE=DATA,RFILE=FFFFFFF(,DELETE=PPPPHH(,FAST=YES)(,NRECDS=NN)
(,CYLOFLO=NN)(,DT=AABBBCCCDDDDDEEEEE)
```

Copy Data File (ISAM):

```
SAVE=DATA,RESTORE=DATA,SFILE=FFFFFFF,RFILE=FFFFFFF
(,DELETE=PPPPHH)(,NRECDS=NN) (,CYLOFLO=NN)(,FAST=YES)
(, COPY=DATA )
```

Non VSE-SP Library Functions

The following section discusses Core Image, Relocatable, Source Statement, Procedure, and DOS Private libraries.

A system library may be restored as a private library and vice versa. In other words, once a library is saved or copied it loses any attributes regarding private or system per se). Specifying the type of library to be built reassigns these attributes. Also, the restored library is independent of the saved library in the following parameters: size, location on disk, and types of disks.

Upon restoration, the library is automatically condensed. Also, you may tell Dr.D to specifically include or exclude certain members from the library when restoring (see Section "Member Selection" on page 168 for more information on 'MEMBER=SELECT/REJECT').

Since a restore condenses a library, Dr.D may be used as a high speed library backup and condense system.

***Warning* NEVER RESTORE A CORE IMAGE LIBRARY TO THE SYSRES YOU ARE CURRENTLY IPLed FROM**

Saving, Restoring, Copying System Libraries

How to Save a System Library

Specifying 'SAVE/COPY=SC, SR, SS, SP' will SAVE/COPY a system CORE, RELO, SOURCE, OR PROC library, respectively.

The SYSRES on which these libraries reside is assumed to be on the drive where SYSRES is assigned. This may be overridden by specifying an alternate: 'SSYSNO=SYSXXX'. Thus you may save a system library from which a SYSRES is not IPLed.

Example 9. Save Source Statement Library from SYSRES.

This example saves the system source statement library from SYSRES, and then from a non-resident SYSRES pack on 131.

```
// JOB L9
// ASSGN SYS007,280
// TLBL TAPEOUT,'SSL'
// ASSGN SYS015,131
// EXEC DRD,SIZE=200K
   SAVE=SSOURCE
   SAVE=SSOURCE,SSYSNO=SYS015
/*
/&
```

Restoring A Library as a System Library

Specifying 'COPY/RESTORE=SC, SR, SS, SP' will COPY/RESTORE a system CORE, RELO, SOURCE, or PROC library, respectively.

In order to restore a system library, you must have a 'SYSRES' pack (resident or non-resident) to restore to.

Specifying 'LIB=O' (OLD) will restore the library on top of the old corresponding library on the SYSRES pack, using the same library size and number of directory tracks.

Specifying 'LIB=A' (ADD) will add the saved library to the old corresponding library on the SYSRES pack. The default is 'LIB=O'.

If you are restoring a system core image library, Dr.D assumes it will reside on the pack pointed to by SYS002.

If you are restoring any other type of system library, Dr.D will assume the target disk to be pointed to by SYSRES. Having the default of SYS002 for system core will reduce the chances of accidentally corrupting SYSRES. You may override these assumed default 'SYS' numbers by specifying a 'RSYSNO=SYSXXX' on the control card.

Example 10. Save System Relocatable library.

This example saves a system relocatable library.

```
// JOB L10 SAVE SYSTEM RELO
// ASSGN SYS007,280
// TLBL TAPEOUT,'MY RELO'
// EXEC DRD,SIZE=200K
   SAVE=SRELO
/*
/&
```

Example 11. Restore System Relocatable library.

This example restores a system relocatable library.

```
// JOB L11
// ASSGN SYS008,280
// TLBL TAPEIN,'MY RELO'
// EXEC DRD,SIZE=200K
   RESTORE=SRELO
/*
/&
```

Note: That the preceding two examples have also created a back-up of the relocatable library and condensed it.

Example 12. Merge 3380 System Source library to 3350 pack.

This example assumes a system is IPLed from a 3380 SYSRES. A non- resident 3350 SYSRES pack exists on 131. The job merges the 3380 system source statement library to the library on the 3350 pack.

```
// JOB L12
// ASSGN SYS004,131
// EXEC DRD,SIZE=200K
   COPY=SSOURCE,RSYSNO=SYS004,LIB=ADD
/*
/&
```

Summary of Control Card Format for System Libraries

Save a System Library

```
SAVE=SCORE(,SSYSNO=SYSXXX)
     SRELO
     SSOURCE
     SPROC
```

Restore a System Library

```
RESTORE=SCORE(,RSYSNO=SYSXXX)(,LIBRARY=OLD)
        SRELO    (,LIBRARY=ADD)
        SSOURCE
        SPROC
        PPROC
```

Copy a System Library

```
SA=SCORE,RE=SCORE(,SSYSNO=SYSXXX)(,RSYSNO=SYSXXX)(,LIBRARY=OLD)
(  COPY=SCOR   )                          (,LIBRARY=ADD)
SAVE=SRELO,RESTORE=SRELO
(  COPY=SRELO  )
SAVE=SSOURCE,RESTORE=SSOURCE
( COPY=SSOURCE )
SAVE=SPROC,RESTORE=SPROC
(  COPY=SPROC  )
```

Default:

```
SSYSNO=SYSRES
RSYSNO=SYSRES        EXCEPT FOR SYSTEM CORE
RSYSNO=SYS002        FOR SYSTEM CORE
LIBRARY=OLD          IS ASSUMED UNLESS OTHERWISE SPECIFIED
```

Saving, Restoring, Copying SYSRES

Example 13. Save a SYSRES.

This example saves a SYSRES that has a CORE and RELO library on it. It then creates a condensed version of the SYSRES on a work pack on 131.

```
// JOB L13
// ASSGN SYS007,280
// TLBL TAPEOUT,'MY SYSRES'
// EXEC DRD,SIZE=200K
   SAVE=SCORE
   SAVE=SRELO
/*
/&
```

Example 14. Restore SYSRES.

This example creates a new SYSRES pack with library functions on it- use the IBM CORGZ program to create an empty sysres before doing the restore.

```
// JOB L14
// ASSGN SYS008,280
// TLBL TAPEIN,'MY SYSRES'
// ASSGN SYS005,131
// DLBL NEWSYS,'SYSRES PACK REL 00',99/365,SD,CISIZE=512(FBA)
// EXTENT SYS005,111111,1,0,1,3979   CKD
// EXTENT SYS005,111111,1,0,2,3978   FBA
// EXEC DRD,SIZE=200K
   RESTORE=SCORE,RSYSNO=SYS005
   RESTORE=SRELO,RSYSNO=SYS005
/*
/&
```

Example 15. Condense without creating backup tape.

```
// JOB L15B EASY
// ASSIGN SYS005,131
// EXEC DRD,SIZE=200K
   COPY=CURRENT,SSYSNO=SYSRES,RSYSNO=SYS005
   COPY=SCORE,RSYSNO=SYS005
/*
/&
```

Saving Private Libraries (VSE I)

Specifying 'SAVE/COPY=PC,PR,PS,PP' will SAVE/COPY a 'Private-CORE, RELO,SOURCE, or PROC Library', respectively. Dr.D will find the particular library by opening the files: 'IJSYSCL', 'IJSYSRL', 'IJSYSSL', 'IJSYSPL', respectively.

This may be overridden by supplying DLBL/EXTENTS and specifying 'SFILENAME='.

Example 16. Save a Private Source library.

This example saves a private source statement library.

```
// JOB L16
// ASSGN SYS007,280
// TLBL TAPEOUT,'SSL SAVE'
// DLBL IJSYSSL,'SOURCE',99/360,SD
// EXTENT SYSSLB,123456,1,0,20,1000
// DLBL MYSORCE,'SOURCE',99/360,SD
// EXTENT SYS020,123456,1,0,20,1000
// ASSGN SYSSLB,131
// ASSGN SYS020,131
// EXEC DRD,SIZE=200K
   SAVE=PSOURCELIB
   SAVE=PSOURCELIB,SFILENAME=MYSORCE
/*
/&
```

Sometimes due to the way a system is configured, it is difficult to supply label information for a library. An alternative way of telling Dr.D where the library is, is to specify 'SIDENT='. Dr.D will find the library on SYSCLB, SYSRLB, or SYSSLB (corresponding to SAVE/COPY=PC, or PS) that has a file identification equal to the one specified in 'SIDENT='. Further, the default SYS-numbers (SYSCLB, SYSRLB, SYSSLB, SYSPLB) may be overridden by specifying 'SSYSNO=SYSXXX'.

Example 17. Save a Private Source library.

This example provides another way to save the source statement library in Example 16.

```
// JOB L17
// ASSGN SYS007,280
// TLBL TAPEOUT,'PSL SAVE'
// ASSGN SYSSLB,131
// ASSGN SYS020,131
// EXEC DRD,SIZE=200K
   SAVE=PSOURCELIB,SIDENT=SOURCE
   SAVE=PSOURCELIB,SIDENT=SOURCE,SSYSNO=SYS020
/*
/&
```

Summary of Control Card Format for Saving a Private Library (VSE I)

Save a Private Library using DLBL/EXTENTS:

```
SAVE=PCORE(,SFILE=XXXXXXX)
SAVE=PRELO
SAVE=PSOURCE
SAVE=PPROC
```

Default:

IF 'SF=' IS NOT SPECIFIED Dr.D WILL OPEN THESE FILENAMES INSTEAD-

FOR SAVE=PCORE,	IJSYSCL
FOR SAVE=PRELO,	IJSYSRL
FOR SAVE=PSOURCE,	IJSYSSL
FOR SAVE=PPROC,	IJSYSPL

Save a Private Library Using FILE-ID and SYSNO:

```
SAVE=PCORE,SIDENT=XXX...XXX(,SSYSNO=SYSXXX)
SAVE=PRELO
SAVE=PSOURCE
SAVE=PPROC
```

Default:

IF 'SS=' IS NOT SPECIFIED Dr.D WILL ASSUME THE FILE-ID ENTERED IN 'SI=' WILL BE FOUND ON THE PACK POINTED TO BY THE FOLLOWING SYSNO-

FOR SAVE=PCORE,	SYSCLB
FOR SAVE=PRELO,	SYSRLB
FOR SAVE=PSOURCE,	SYSSLB
FOR SAVE=PPROC,	SYSPLB

Restoring Private Libraries

Specifying 'COPY/RESTORE=PC, PR, PS, PP' will rebuild a Private- CORE, RELO, SOURCE, or PROC library, respectively.

The library may be restored on top of an existing library, completely overlaying its contents but having the same size and number of directory tracks. This is done by putting 'LIB=O' (OLD).

The library may be restored adding its contents to a library. This is done by setting 'LIB=A' (ADD). In either case the restored library already exists and may be communicated to Dr.D by any of the four optional ways shown below.

1. Default filename (IJSYSCL, IJSYSRL, IJSYSSL, IJSYSPL)
2. Specified filename ('RFILE=').
3. Specified file-ID ('RIDENT=') with default SYSNO (SYSCLB, SYSRLB, SYSSLB, SYSPLB).
4. Specified file-ID ('RIDENT=') and specified SYSNO ('RSYSNO=') the default is 'LIB=OLD'.

Example 18. Backup, Condense, Private Source library.

This example condenses a Private Source statement library on 131. Assume SYSSLB is permanently assigned to 131. Also assume DLBL/EXTENTS for IJSYSSL are in the label area.

```
// JOB L18 BACK-UP PSL
// ASSGN SYS007,280
// TLBL TAPEOUT,'SAVE PSL'
// EXEC DRD,SIZE=200K
   SAVE=PSOURCELIB
/*
/&

// JOB L18 CONDENSE PSL
// ASSGN SYS008,280
// TLBL TAPEIN,'SAVE PSL'
// EXEC DRD,SIZE=200K
   RESTORE=PSOURCELIB
/*
/&
```

Sometimes it is necessary to change the size of the directory, or library, or create a private library on a pack where there wasn't one before. To accomplish this specify 'LIB=N' (NEW). It will create and format an empty private library before Copying/Restoring. The library will have a total size equal to that specified in the DLBL and extents defining the library. The rebuilt library will have a number of directory blocks or tracks in exact proportion to the ratio of directory size to the total size of the original library (i.e., if the original library has 100 cylinders and 10 directory tracks and the rebuilt one has 150 cylinders, it will have fifteen

directory tracks). This may be overridden by explicitly specifying 'DIRTRKS=NN' or 'NNN'(FBA).

For 'LIB=NEW' only two methods are available to specify to Dr.D where the new library is to be located.

1. Default filenames- DLBL/EXTENTS for IJSYSCL, IJSYSRL, IJSYSSL and IJSYSPL are supplied in JCL or standard labels and Dr.D will open the file IJSYSCL, IJSYSRL, IJSYSSL or IJSYSPL for 'COPY/RESTORE=PC, PR, PS, PP' respectively.
2. Explicit filenames - DLBL/EXTENTS for the library are supplied and the filename is specified in 'RF='.

Example 19. Create Private Relocatable library.

This example assumes a tape, a 3350 with a RELO library of 50 cylinders and 20 directory tracks. It creates a private relo library on a 3380 with 100 cylinders and 19 directory tracks. Two methods of achieving this goal are shown.

```
// JOB L19 ONE
// ASSGN SYS008,280
// TLBL TAPEIN,'RELO'
// ASSGN SYSRLB,1C1
// DLBL IJSYSRL,'IJSYSRL',99/360,SD
// EXTENT SYSRLB,222222,1,0,19,1900
// EXEC DRD,SIZE=200K
   RESTORE=PRELOLIB,DIRTRKS=19,LIBRARY=NEW
/*
/&

// JOB L19 TWO
// ASSGN SYS008,280
// TLBL TAPEIN,'RELO'
// ASSGN SYS013,1C1
// DLBL AFILE,'IJSYSRL',99/360,SD
// EXTENT SYS013,222222,1,0,19,1900
// EXEC DRD,SIZE=200K
   RESTORE=PRELO,DIRTRKS=19,LIBRARY=NEW,RFILENAME=AFILE
/*
/&
```

Note: It is not a good idea for any other partition to be accessing a private library when it is being restored. Nor is it beneficial when restoring a Private Core Image library to have SYSCLB assigned to it in any partition (including the one assigned to Dr.D).

Example 20. Save and Restore Private Core library.

This example creates three job streams: The first saves a Private Core Library. The second restores it condensed. The third increases the directory tracks to 20.

Assume: / / DLBL IJSYSCL,'PVT CORE',99/360,SD
/ / EXTENT SYSCLB,111112,1,0,20,1000
are in the standard label area and that:
ASSGN SYS007,280
ASSGN SYS008,280
ASSGN SYSCLB,132
are permanent assignments.

```
// JOB L20 ONE
// TLBL TAPEOUT,'SAVE CORE'
// EXEC DRD,SIZE=200K
   SAVE=PCORE
/*
/&

// JOB L20 TWO
// TLBL TAPEIN,'SAVE CORE'
// ASSGN SYS020,132
// ASSGN SYSCLB,UA
// EXEC DRD,SIZE=200K
   RESTORE=PCORE,RIDENT='PVT CORE',RSYSNO=SYS020
/*
// ASSGN SYSCLB,132
/&

// JOB L20 THREE
// TLBL TAPEIN,'SAVE CORE'
// ASSGN SYS020,132
// ASSGN SYSCLB,UA
// DLBL IJSYSCL,'PVT CORE',99/360,SD
// EXTENT SYS020,111112,1,0,20,1000
// EXEC DRD,SIZE=200K
   RESTORE=PCORE,DIRTRKS=20,LIBRARY=NEW
/*
/&
```

Remember that a library may be restored/copied as a system or private library independent of what it was when saved.

Example 22. Save a System Relocatable library.

This example saves a system relocatable library on a SYSRES pack from which the system was not IPLed. It then restores this tape, creating a 3350 private library with and 50 cylinders and 20 directory tracks on a scratch pack on 133.

```
// JOB L22 SAVE
// ASSGN SYS007,280
// TLBL TAPEOUT,'RELO'
// ASSGN SYS012,132
// EXEC DRD,SIZE=200K
   SAVE=SRELO,SSYSNO=SYS012
/*
/&

// JOB L23 RESTORE
// ASSGN SYS008,280
// TLBL TAPEIN,'RELO'
// DLBL IJSYSRL,'RELO LIB',99/360,SD
// EXTENT SYS003,121212,1,0,20,1000
// ASSGN SYS003,133
// EXEC DRD,SIZE=200K
   RESTORE=PRELO,LIB=NEW,DIRTRKS=20
/*
/&
```

Summary of Control Card Format for Restoring a Private Library

Restore Private Library Using DLBL/EXTENTS:

```
RESTORE=PCORE(,RFILE=XXXXXXX)(,LIBRARY=OLD)
RESTORE=PRELO                (,LIBRARY=NEW)(,DIRTRKS=NN)
RESTORE=PSOURCE              (,LIBRART=ADD)
RESTORE=PPROC
```

Default:

IF 'RF=' IS NOT SPECIFIED Dr.D WILL OPEN THESE FILENAMES INSTEAD:

FOR RESTORE=PCORE, IJSYSCL
FOR RESTORE=PRELO, IJSYSRL
FOR RESTORE=PSOURCE, IJSYSSL
FOR RESTORE=PPROC, IJSYSPL

IF 'LIBRARY=' IS NOT SPECIFIED Dr.D WILL ASSUME 'LIBRARY=OLD'.

Restore Private Library Using FILE-ID and SYSNO:

```
RESTORE=PCORE,RIDENT=XXX...XXX(,RSYSNO=SYSXXX)(,LIBRARY=OLD)
RESTORE=PRELO                                 (,LIBRARY=ADD)
RESTORE=PSOURCE
RESTORE=PPROC
```

Default:

IF 'RS=' IS NOT SPECIFIED Dr.D WILL ASSUME THE FILE-ID ENTERED IN 'RI=' WILL BE FOUND ON THE PACK POINTED TO BY THE FOLLOWING SYSNO-

FOR SAVE=PCORE, SYSCLB
FOR SAVE=PRELO, SYSRLB
FOR SAVE=PSOURCE, SYSSLB
FOR SAVE=PPROC SYSPLB

IF 'LIBRARY=' IS NOT SPECIFIED Dr.D WILL ASSUME 'LIBRARY=OLD'.

Member Selection

When you are saving, copying, or restoring a library it would be helpful to have the facility for including or excluding certain members. This can be done with Dr.D by specifying 'MEMBER=S' (Select) or 'MEMBER=R' (Reject), respectively on the Dr.D control statement.

A member of a library is either a 'Phase, Module, Book, Macro, or Procedure' depending on whether the library is 'Core, Relo, Source, or Proc'.

Member selection can be used to limit both members read (Saved) and those written (Restored/Copied). You must inform Dr.D of the members to be considered by supplying 'Member' cards (immediately following the library Copy/Restore control card) with the member name punched in the first eight positions. If you're working with a source statement library, the sub-library goes into position one of the card and the book name goes into positions two to nine.

The '.' is omitted. Overlapping or subset cards may be entered to allow some members of a Selected/Rejected group to be treated in the opposite manner of the rest of the group (see example 27). A '/*' denotes the end of the 'member' cards. Do not confuse this '/*' with the one denoting the end of input for Dr.D. If 'MEMBER=' is specified you need them both.

Example 23. Selective restore.

This example restores the relocatable library as in example 22, but does not restore module 'IJKL4'.

```
// JOB L23 RESTORE
// ASSGN SYS008,280
// TLBL TAPEIN,'RELO'
// DLBL IJSYSRL,'RELO LIB',99/360,SD
// EXTENT ,121212,1,0,20,1000
// EXEC DRD,SIZE=200K
   RESTORE=PRELO,LIBRARY=NEW,DIRTRKS=20,MEMBER=REJECT
   IJKL4
/*   (END OF MEMBER CARDS)
/*   (END OF Dr.D CONTROL CARDS)
/&
```

Example 24. Merge Member.

This example retrieves an accidentally corrupted book on the System Source statement library (named 'A.PROG1234') from the save tape. It then merges the uncorrupted version with the current library.

```
// JOB L24 MERGE MEMBER
// ASSGN SYS008,280
// TLBL TAPEIN,'SOURCE SAVE'
// EXEC DRD,SIZE=200K
   RESTORE=SSOURCE,LIB=ADD,MEMBER=SELECT
   A.PROG1234
/*
/*
/&
```

A '/' in any position of a member control card removes that position in the name from consideration in the selection or rejection process. To select or reject all library members beginning with IJK from a save tape, supply the member card:

IJK/ / / / /

Likewise, to select or reject all members with an '8' in the fifth position of the name:

/ / / /8/ / /

Note: This important distinction: a card such as: ABCD will only select or reject the member 'ABCD' while one such as: ABCD/ / / / will select or reject not only the member 'ABCD' but also any of its cousins such as 'ABCD0001', 'ABCDxxxx', etc.

Example 25. Save Core Image library.

This example creates a Private Core Image library, whose only contents are the thirteen phases and three transients of the ANSI COBOL COMPILER (360N-CB-482). Following are two alternative job streams to accomplish the same goal.

```
// JOB L25 LONG
// ASSGN SYS008,280
// TLBL TAPEIN,'CORE LIB'
// DLBL IJSYSCL,'PCIL',99/360,SD
// EXTENT ,111111,1,0,20,1000
// EXEC DRD,SIZE=200K
   RESTORE=PCORE,LIB=NEW,DIRTRKS=20,MEMBER=SELECT
   FCOBOL
   FCOBOL11
   FCOBOL12
   FCOBOL20
   FCOBOL21
   FCOBOL22
   FCOBOL30
   FCOBOL40
   FCOBOL50
   FCOBOL51
   FCOBOL60
   FCOBOL61
   FCOBOL70
   $$BCOBER
   $$BCOBR1
   $$BFCMUL
   /*
/*
/&
```

```
// JOB L25 SHORT
// ASSGN SYS008,280
// TLBL TAPEIN,'CORE LIB'
// DLBL IJSYSCL,'PCIL',99/360,SD
// EXTENT ,111111,1,0,20,1000
// EXEC DRD,SIZE=200K
   RESTORE=PCORE,LIB=NEW,DIRTRKS=20,MEMBER=SELECT
   FCOBOL//
   $$BCOB//
   $$BFCMUL
   /*
/*
/&
```

Example 26. Rebuild Private Core Image Library.

This example rebuilds the Private Core Image library as in example 25, restoring all the phases except the FCOBOL ones.

```
// JOB L26
// ASSGN SYS008,280
// TLBL TAPEIN,'CORE LIB'
// DLBL IJSYSCL,'PCIL',99/360,SD
// EXTENT ,111111,1,0,20,1000
// EXEC DRD,SIZE=200K
   RESTORE=PCORE,LIB=NEW,DIRTRKS=20,MEMBER=REJECT
   FCOBOL//
   /*
/*
/&
```

Example 27. Rebuild Private Core Image library.

Same as above, except this example adds the FCOBOL members to USRCL1 and excludes FCOBOL50.

```
// JOB L27
// ASSGN SYS008,280
// TLBL TAPEIN,'CORE LIB'
// DLBL USRCL1,'USER PCL'
// LIBDEF CL,TO=USRCL1
// EXEC DRD,SIZE=200K
   RESTORE=PCORE,LIB=ADD,MEMBER=SELECT
   FCOBOL//
   FCOBOL50   @REVERSE@
   /*
/*
/&
```

Summary of Control Card Format for Member Select.

Select or Reject Members of a Library:

```
RESTORE=XX(,...ETC...),MEMBER=SELECT (OR MEMBER=REJECT)
COPY=XX  Where XX is either SC,SR,SS,SP,PC,PR,PS,PP
```

Appendix I - Summary of Dr.D Keywords

To further illustrate the operation of Dr.D operands, the following is a summary of all keywords and their allowable operand values.

Global Keywords Note: That global keywords, once entered, remain in effect until changed (or execution terminates). Global keywords are indicated by '(GLOBAL)'. We recommend that Global operands be specified on 'NOP' control statements preceding SAVE/RESTORE/COPY statements.

`ABUFF=nnn`

This keyword is used to specify the buffer size for the VSAM BUILD AIX function, where nnn is a one to three digit specifying the buffer size in thousands (K).

`AFILE=nn/ddddddd`

This keyword is used with the VSAM alternate index BUILD feature to specify either the DD/DLBL name of the AIX or the number of AIX files to be created. If the value entered is a one or two digit numeric it is assumed to be the number of AIX files to be created, otherwise it is assumed to be the DD/DLBL name.

`ALIGN=YES/NO/ALL`

This operand is used during a reorganizing restore (CKD disk only) of a 'SAVE=CURRENT' volume backup, to specify how files are to be cylinder aligned. The assumed value is 'YES' indicating that files that were cylinder aligned will remain aligned. If 'NO' is entered, no files within the reorg area will be cylinder aligned. If 'ALL' is entered, each file within the reorg area of the disk will be aligned.

`ALLOCS=NN  (Do Not Abbreviate)`

This operand is used with the SAVE=LOGICAL (VSAM) backup to select files based on the number of allocations for each file. It is used as a 'selector card' in conjunction with PARTIAL=SELECT/REJECT and causes files to be selected (not rejected) if they have allocations equal or greater than that entered. If ALLOCS=0 is entered, only files having no allocations are selected.

```
AREA=SELECT/REJECT/COMPACT
or
AREA=(S1-E1,S2-E2,...)
```

For use during restore/copy of data areas by ALL,CURRENT, or PHYSICAL save methods. This allows restore to be selective down to the track or FBA block level. Like library member selection, control cards are entered to define the areas to be restored to. As many Select/Reject cards may be entered as will fit in partition memory and have the following format:

```
000000-0001111
123456-7890123
BBBBBB-BBBBBBX      (for FBA disk)
```

```
CCCHH-CCCHH  X      (for CKD disk)
```

In each case, beginning and ending disk address is specified as either FBA block number or CKD cylinder and head(track). Column 13 is optional and may be used to override the 'AREA=SE/RE' specified (for that control card only).

Overlapping specification is allowed with subsets processed first, for example:

```
RE=PH,RB=000001,RH=100000,AR=SE
1-1000
11-11R
/*
```

This would restore FBA blocks to 1-10 and 12-1000 with block 11 rejected. The same results from:

```
RE=PH,RB=000001,RH=100000,AR=SE
1-10
12-1000
/*
```

Relocation of disk areas is possible with use of 'RB' and 'RH' parameters. Area selection is based on the addresses restored to, not the addresses saved from. In the example above, if the area being saved from was 100001-200000 then the blocks restored to 1-10 and 12-1000 would have originally been located at 100001-100010 and 100012-101000.

`AREA=(S1-E1,S2-E2,...)`

This form of the AREA parameter is used to define the area(s) of disk to be saved from or restored to during SAVE=LOGICAL (VTOC) processing. Each set of parameters is a starting and ending track or FBA block number separated by a dash '-'. If the specification is a repeat of the last entered, a single 'R' may be entered. The entries are assumed to be in one to one correspondence with the 'SV=' entries if saving or the 'RV=' entries if restoring.

`ASSOC=YES/NO`

When Saving/Restoring VSAM with SAVE/RESTORE=LOGICAL, indicates whether associated files are to be Saved/Restored.

Note: That this does not apply to SAVE=VSAM type backups, only those produced by SAVE=LOGICAL.

`AUTO=YES/NO/OVERRIDE/ROTATE`

For VSAM, indicates whether automatic file definition is to be invoked, and to allow the definition to be altered by user override statements. 'ROTATE' is used to cause file and file component(s) to be 'rotated' among the candidate volumes of the target catalog, allowing the index component to be on a different volume from the data component and files to be more evenly spread over the candidates.

`BLOCKSIZE=NNNNN` `(GLOBAL)`

'BLOCKSIZE=NNNNN' where 'NNNNN' is a five digit decimal number representing the Dr.D blocksize. It may be specified during Saves, Restores, or

Copies. It may be smaller than the default value (useful in sending tapes to smaller installations) or larger (faster execution times and less tape used). It may not be set to a value less than the largest disk block (+16) in the disk area being saved, nor may it exceed 65496. Once a blocksize has been set it is in effect for the remainder of the Dr.D execution or until another 'BL=' operand is submitted. When used with the Data Tape (DT=YES) option, 'NNNNN' specifies the Blocksize of the tape record.

It must be less than 65497 and at least as large as the logical record length plus nine. In no event will other than full logical records be written to tape. If used during copy operations, memory requirements, performance and paging rates will be affected in most cases.

NOTE: This parameter should not be used in conjunction with 'FIND=' 'SKIP=', or 'POSITION=. Instead, set the blocksize in a preceding 'RE=NOP' control card.

`BUFFERSPACE=NNNNNN` (GLOBAL)

The 'BUFFERSPACE=' keyword is used to override the buffer space for each VSAM file being defined by the Dr.D auto-define feature. If a number greater than 256 is entered, it is used as the buffer space for each file. If it is less than 257, it is used to multiply each file's existing buffer space. If 'BU=1' is entered then no buffer space will be generated during the restore when a delete/ define takes place for a VSAM file. This is a global parameter, and remains in effect throughout the execution or until re-entered (enter 'BU=N' to discontinue its use).

`CARDCOUNT=XX`

'CARDCOUNT=XX' where XX is a two digit decimal number (leading zero) representing a count of Saves, Copies, and Restores. While this operand, if specified, does not have to appear on the first control card, it can't appear more than once. It represents a count of control cards including the one it appears on until the last one that should be in the job stream. This count excludes continuation cards, '$$$$$$' cards, and the '/*' card. If the specified count is either exceeded or not fulfilled, the operator is informed.

`CASPLIT=NN`

The CASPLIT=NN keyword is used with the SAVE=LOGICAL (VSAM) backup in conjunction with PARTIAL=SELECT/REJECT to select files having a number of CA splits equal to or greater than the number specified. This keyword may not be abbreviated and is entered as a file 'selector' card. (See also CISPLIT=NN).

`CBEGIN=CCCCCCHH/BBBBBBBB`

The 'CCCCCC' is a zero to six digit decimal number representing a cylinder and 'HH' is a two digit decimal number (leading zero) representing a head. It is used with COPY=PHYSICAL. If the starting tracks for the save and restore extents are equal, 'SBEGIN' and 'RBEGIN=' may be replaced by 'CBEGIN='.

For FBA disks, this becomes a one to eight decimal digit field representing the starting block number.

`CDATA=PRIMARYCLASS/SECONDARYCLASS (GLOBAL)`

The 'CD=PCLASS/SCLASS' keyword is used to override the data allocation classes of VSAM files being defined by the Dr.D auto- define feature (see also CX=). This is a global parameter and stays in effect throughout the current execution of Dr.D or until changed by a new entry (enter 'CD=N' to discontinue its effect).

`CFACTOR=NNNN`

CFACTOR=NNNN may be specified, when RESTORE=LOGICAL is used, to increase or decrease the space occupied by a file. The 'NNNN' is a percentage or multiplying factor of the form 'NN.NN' so that 'CF=110' would increase the file allocation by 10% and 'CF=90' would decrease by 10%.

`CHALT=CCCCCCHH/BBBBBBBB`

The 'CCCCCC' is a zero to six digit decimal number representing a cylinder and 'HH' is a two digit decimal number (leading zero) representing a head. It is used with 'COPY=PHY' or 'COPY=ALL'. If the starting tracks for the save and restore extents are equal, 'SHALT' and 'RHALT' may be replaced by 'CHALT'.

For FBA disks, this becomes a one to eight decimal digit field representing the ending block number.

`CISPLIT=NN`

The 'CISPLIT=NN' keyword is used with the 'SAVE=LOGICAL' (VSAM) backup in conjunction with 'PARTIAL=SELECT/REJECT' to select files having a number of CI splits equal to or greater than the number specified. This keyword may not be abbreviated and is entered as a file 'selector' card. (See also CASPLIT=NN).

`CK=YES/NO(yes assumed)`

'CK=NO' is used to ignore CKD disk count area errors during backup of files or volumes (such as VM Mini-Disks) where the count area may not reflect the true location of the physical blocks, preventing issuance of message 76.

`CLIBRARY=CCCTT`

'CLIBRARY=CCCTT' where 'CCC' is a three digit decimal number (leading zeros) representing the number of cylinders to be allocated to a SYSTEM CORE IMAGE library and 'TT' is a two digit decimal number (leading zero) representing the number of tracks for that SYSTEM library's directory. This operand is only specified on 'COPY/ RESTORE=REORG'. 'CLIBRARY=' specifies the allocation for the new SYSRES library. Specifying 'S' (same) in place of the 'CCCTT' operand causes Dr.D to use the saved library's allocation. In any case, 'CL=' must be specified, since you can't build a SYSRES without a core image library.

For FBA disks, this becomes a nine digit number consisting of a six digit number as the total library size (in blocks) and a three digit directory block field.

`CN=DATA/INDEX (CISIZE=DATA/INDEX) (GLOBAL)`

The 'CN=DATA/INDEX' keyword is used to override the cisize(s) of VSAM files being defined by the Dr.D VSAM auto definition feature. If a number greater than 256 is entered, it is used to replace each file's DATA/INDEX cisize. If less than 257, it is used to multiply each file's existing cisize. This is a global parameter that remains in effect throughout the current execution or until a new value is entered (enter 'N' to terminate its effect). If both data and index cisizes are entered, they are separated by the '/' (CN=4096/512 for instance).

`COPY=XX`

To make the program operate in the copy mode (Disk to Disk), it is only necessary to put both the 'SAVE=XX' and 'RESTORE=XX' keyword pairs on the same card. In most cases the operand of both keywords are identical and can then be replaced by 'COPY=XX'. One exception is in copying a SYSTEM to PRIVATE (or vice versa) library.

Here, 'SAVE=XX' and 'RESTORE=YY' must be specified to denote the differing type of libraries. See "Appendix VIII - Table of Save, Restore, or Copy Commands" on page 207.

`CP=YES/NO (TAPE COMPRESSION) (GLOBAL)`

If 'CP=YES' is specified during backup to tape, data compression is used to reduce tape usage and channel access time. Once invoked, compression is in effect for the duration of the execution, or until 'CP=NO' is entered.

This feature will increase backup speed while reducing tape and channel usage provided enough CPU power is available. Files containing few repeating characters, or already compressed (Source or Relo libraries for instance) will not benefit from this feature. This parameter is not required for restore.

`CSYSNO=SYSXXX`

'CSYSNO=SYSXXX' where 'SYSXXX' is any valid SYSTEM number (see 'SSYSNO='). When copying from one portion of a disk to another (i.e. SSYSNO and RSYSNO point to the same disk), this operand may be specified in place of the SSYSNO and RSYSNO operands. This is applicable to copies: 'PC, PS, PR, PP, SC, SR, SP, SS, and PH'.

`CTLOWER=XX`

'CT=XX' where 'XX' is a two digit decimal number (leading zero) representing a lower head limit. It may be used to replace both 'ST=' and 'RT=' parameters when copying a physical extent.

`CUPPER=XX`

'CU=XX' where 'XX' is a two digit decimal number (leading zero) representing an upper head limit. It may be used to replace both 'SU=' and 'RU=' parameters when copying a physical extent.

`CVOLID=XXXXXX`

'CVOLID=XXXXXX' where 'XXXXXX' is six alphanumeric characters representing a volume serial number. If 'SVOLID=' and 'RVOLID=' are both specified and are equal, the two of them may be replaced with the single operand 'CVOLID='.

`CX=PRIMARYCLASS/SECONDARYCLASS (GLOBAL)`

The 'CX=PCLASS/SCLASS' keyword is used to override the index allocation classes of VSAM files being defined by the Dr.D auto- define feature (see also CD=). This is a global parameter and stays in effect throughout the current execution of Dr.D or until changed by a new entry (enter CX=N to discontinue its effect).

`CYLOFLO=XX`

'CYLOFLO=XX' where 'XX' is a two digit decimal number (leading zero) denoting the number of tracks per cylinder to be allocated to cylinder overflow. If omitted, the number of CYLOFLO tracks for the saved file is used. If no CYLOFLO tracks are desired, set 'CYLOFLO=00'. When 'CY=XX' is used in conjunction with a volume backup the 'CY=' parameter limits the number of tracks read during each I/O.

`DATE=XXXXXXX`

The DATE= keyword is used to set the expiration date for VSAM and non-VSAM files during file restore or the RESTORE=EXPIRATION date function. The entered field is usually a five digit number of the form 'YYDDD' as a Julian date, but may also take several other formats/options. For VSAM files, specifying DATE=0 causes no 'TO' date to be generated when IDCAMS file definition is invoked by Dr.D. For non-VSAM files, using the RESTORE=EXPIRATION function, the entered date can be prefixed with 'C' or 'A' if a creation or last access date is to be set.

`DCONTROL=`

The 'DC=' (Dr.D Control) parameter is used to pass additional control parameters to Dr.D during the SAVE/RESTORE operation. The information within the parentheses is appended to the generated control card and is not checked until the SAVE/RESTORE begins.

`DD=DDNAME`

The 'DD=' parameter can be used whenever file selection by file- id is desired, but the file id is to be obtained from a DLBL. The DLBL name specified is used to find the DLBL from the label area, and may be a standard or user label. The 'DD=' must be coded with the DLBL/DTF name as the keyword operand. This parameter is coded as a 'selector' card with 'PARTIAL=SELECT/REJECT', and not on the 'SAVE/RESTORE=LOGICAL' control card.

`DELETE=YES`

'DELETE=YES' is used to cause the 'RESTORE=LOGICAL' (VTOC) function to change the expiration date of files about to be restored, so that no overlap messages are produced, and the operator is not required to enter 'DELETE' as each file is restored.

Note: That the files on disk are not deleted, but the 'RESTORE=EXPIRATIONDATE' function is invoked to alter their expiration dates to '00000'.

`DELETE=PPPPXX`

'DELETE=PPPPXX' where 'PPPP' is a four digit decimal number (leading zero) less than 4097, denoting the position in a record (relative to one) of a one byte delete code and 'XX' is a two character field representing that delete code in hexadecimal. Each 'X' may be one of the following codes: '0123456789ABCDEF'.

This operand is only used on 'COPY/RESTORE=DATA/VSAM/ISAM/PSAM' for VSAM/ISAM/PSAM files. If omitted, all records will be restored.

`DIRTRKS=XX(CKD)/XXX(FBA)`

'DIRTRKS=XX' where 'XX' is a two/three digit (leading zero if necessary) decimal number. This operand may only be specified when 'COPY/RESTORE=PR, PC, PS, PP, and LI=N'. When restoring a private library (PR, PC, PS, PP) and you are creating a library on a pack (LI=N - 'CORGZ' step) the total size of the library is exactly the number of tracks/blocks specified on the DLBL/EXTENT cards for the library. The number of tracks/blocks for the directory of that library is computed by making it in the same ratio to total library blocks as the saved library. This value may be overridden by specifying 'DIRTRKS=XX/XXX'.

`DM=S/E/D`

This parameter is used during 'RESTORE=LOGICAL' (VTOC) processing to specify the name of a 'Disk Manager' in effect to allocate space for the restore process. If 'DM=S' is entered, a sysin file is produced with Space Manager JCL, otherwise DLBL and extent entries are in normal VSE format with a constant starting track or FBA block number. If 'DM=E' is entered, the disk manager is EPIC, while 'DM=D' is used to specify DYNAM/D.

`DTAPE=YES`

'DTAPE=YES' is specified when saving/restoring data files in data-tape format. The DTAPE parameter may be used for multiple file saves in a single execution, producing multiple standard labeled tape files (stacked or single).

'DTAPE=AABBBCCCDDDDDEEEEE' is specified when restoring VSAM or ISAM files from a data tape. Parameter values are:

AA	=	NO. Tracks cylinder overflow ('00' VSAM)
BBB	=	NO. Logical records/physical block ('000' VSAM)
CCC	=	Key length

DDDDD = Logical record length
EEEEE = Starting position of key (counting from 1)

`EB=XXXXX`

'EB=XXXXX' (extent begin) is used during volume logical restore to set the relative track starting address for all generated extent cards. If 'EB=NO' is specified, no extent cards are generated.

If 'EB=RNNNNNN' is specified, extent cards are generated so the first starts at the 'NNNNNN', and each succeeding file is contiguous with the last, allowing relocation and reorganization of disk space.

If 'EB=RS' is specified, an 'inplace' reorganization is indicated where each file stays on the disk(s) it resided on during backup.

'EB=RS' requires the use of the 'AREA=' parameter to define the area(s) of the disk(s) being reorganized.

`EOF=YES/S`

'EOF=YES' is used to indicate that the saving/copying of an extent of a non-VSAM file on a CKD disk is to cease once the hardware EOF is detected. For VSAM files, 'EOF=S' can be used to specify that file saving/copying is to continue until a 'software' EOF is found, without regard to the file's high used RBA (relative byte address).

`ERROR=IGNORE`

This parameter is used in conjunction with 'PARTIAL=' to prevent cancellation when no files are selected for backup/restore. See DOCTOR 171 and DOCTOR 316 error messages.

```
EXITISAABBBCCCDDDDDEEEEE
EXITFBFFFFFHHH
EXITFUFFFFFGGG
```

EXIT... Statements are individual director cards required immediately after an operation director card containing the 'USER=...' parameter.

'EXITISAABBBCCCDDDDDEEEEE' is specified when user-exits are processing ISAM files. Parameter values are:

IS = Indicates ISAM organization
AA = No. Tracks cylinder overflow
BBB = No. Logical records per physical block
CCC = Key length
DDDDD = Logical record length
EEEEE = Starting position of key (counting from 1)

'EXITFBFFFFFHHH' is specified when user-exits are processing fixed blocked SAM or DAM files. Parameter values are:

FB = Indicates fixed blocked
FFFFF = Exact physical blocksize

HHH = No. Logical records per block

'EXITFUFFFFFGGG' is specified when user-exits are processing fixed unblocked SAM or DAM files. Parameter values are:

FU = Indicates fixed unblocked
FFFFF = Exact physical blocksize
GGG = Key length

If a key is present, it must start in position one of the record. Only one exit... Card permitted per operation card. This parameter is invalid for other functions or operations.

`EXPIRED=YES/NO/ONLY/+NNN/-NNN/YYDDD`

This keyword is used when files are to be selected according to their expiration dates. If 'EXPIRED=YES' is specified, files are selected without regard to their expiration dates. If 'NO' is entered, only files unexpired as of the 'IPL' or 'SYSTEM' date are selected. If 'O' is entered, only expired files are selected, and if a date is to be specified, it must be coded as 'O' with the date specification appended (EX=O-10 for instance). If the IPL/SYSTEM date is not to be used to determine whether or not a file is expired, the actual Julian date (YYDDD) can be entered, or the IPL/SYSTEM date can be adjusted forward or backward by entering a '+' or '-' followed by a 1-3 digit number. 'EXPIRED=YES' is assumed for all functions except the 'SAVE/COPY= CURRENT' volume functions where 'EXPIRED=NO' is assumed.

`FAST=YES/NO`

'FAST=YES' may be specified when saving a VSAM or FBA SD file to invoke high speed backup using physical IOCS rather than the slower logical IOCS. The output is identical in either case. If specified with 'SAVE=LOGICAL', it then applies to all files saved. 'FAST=YES' may also be specified when restoring an ISAM file to invoke high speed restore.

`FD=R/D/N/C/M` `(GLOBAL)`

This parameter is used during VSAM restore (AUTO=YES), to specify the method used to delete/define the output file. The assumed value is 'R' meaning that the existing definition is to be reset and reused if the file exists (if it doesn't, it is defined). If 'D' is specified, the file is always deleted and redefined. If 'N' is specified, the file is deleted/defined unless it is reusable or empty. If 'C' is specified, the file is deleted and redefined, with the primary allocation increased so that the restored file will be contained within the primary extent (the secondary size allocation is retained). If 'M' is specified, the primary size is recomputed based on the high-used RBA of the file and its other attributes so that the primary will be the Minimum size needed to contain the data (the secondary size is retained).

```
FILESIZE=NNNNNN
FILESIZE>NNNNNN
FILESIZE<NNNNNN
```

The 'FILESIZE' keyword is used in conjunction with the 'VTOC LOGICAL' functions and 'PARTIAL=SELECT/REJECT' to select non-VSAM files by the total number of tracks/FBA blocks in the extents the non-VSAM file has as defined in the VTOC(s) of the disk(s) being accessed by DR.D. It is entered as a 'selector' card (not as a keyword on the function control card). The functions that use this keyword are BACKUP, LISTVTOC, PUNCH, and DELETE.

```
FILETYPE=K/E/S/R
```

The 'FILETYPE=KESR' keyword is used in conjunction with the 'VSAM LOGICAL' functions and 'PARTIAL=SELECT/REJECT' to select files by their type. It is entered as a 'selector' card (not as a key- word on the function control card), and each position can be one of the following:

K- KSDS file type
E- ESDS file type
S- SAM file type
R- RRDS file type

```
FIND=XXXXXXX or FIND= 'XXX.......XX' or FIND=(SSSSSS)
```

The FIND= function allows tape positioning based on the DLBL name, the file ID, or the disk volume serial number of the file or disk previously saved to tape. If searching is to be based on the 44 character file ID, enclose the ID in quotes ('). If searching is to be based on the volume serial number enclose the volser within parentheses. In searching by volser, only backups created by save all, current, or physical, are examined in the search. The save tape is scanned starting at its current position forward until either the search argument is found or the end of tape occurs. If the user knows which volume of multivolume files to mount, searching may begin with any volume. See message: 'DOCTOR 616' in the message section for additional information.

```
FORMAT=XXXX                                                    (GLOBAL)
```

The 'FORMAT=XXXX' key word is used to control the format of Dr.D reports and log output, and specify tape copy method.

```
**************** LIST DIRECTORY FUNCTION *****************
```

FORMAT=A member list only (no recap)

FORMAT=B no member list, recap only

```
******************** LISTCAT FUNCTION ********************
```

FORMAT=A minimum number of extents is listed

FORMAT=M minimal file information is listed

FORMAT=n 'n' is the maximum number of lines 0 - 9 to print for each file

```
**************** LIST VTOC FUNCTION **********************
```

FORMAT=A file list only (no recap)

FORMAT=B no file list, recap only

FORMAT=X full list (assumed option)

If the 2nd character of the entry is 'E' (XE, AE, BE) then an extent list is produced.

```
******************** TAPE COPY FUNCTION ******************
```

FORMAT=IMAGE tape copied in image mode- output tape is not opened and is exactly like the input tape, including tape volser and label information.

```
***************** LIBRARY SAVE/RESTORE/COPY ***************
               MEMBER NAMES PRINT   SUB-LIBRARY NAMES PRINT
               ON LOG     ON LIST   ON LOG          ON LIST
 FORMAT=A      YES        YES       YES             YES
 FORMAT=B      NO         YES       YES             YES
 FORMAT=C      NO         YES       NO              YES
 FORMAT=D      YES        NO        YES             NO
 FORMAT=E      NO         NO        NO              NO
```

`FREESPACE=XX/XX` (GLOBAL)

The 'FREESPACE=CI/CA' keyword is used to override the VSAM FREESPACE file definition for each file defined by the 'AUTO DEFINITION' feature. the '/' separates the CI freespace from the CA freespace. This remains in effect during the execution of Dr.D unless changed by entering a new value (enter 'FR=N' to terminate this override before end of execution).

`FREESPACE>NN (File condition selector)`

The 'FREESPACE>NN' file condition selector is used to select VSAM files by the % of used space in each file. Unlike the keyword 'FREESPACE' defined above, the file condition selector must be entered like a file ID selector in conjunction with the key word 'PARTIAL='.

`FSECURED=YES`

This parameter specifies that 'SAVE=LOGICAL' (VTOC driven) should save secured files (NO assumed). An operating system message may be issued as each secured file is opened, respond with 'YES' to proceed.

`FUNCTION=XX`

The function command is used in place of 'SAVE/RESTORE/COPY' to invoke certain other functions. At this time the available 'FUNCTIONS' are:

LTAPE - (LISTTAPE) function lists the contents of a backup tape.

LCAT - (LISTCAT) function provides a compact (but complete) report of the objects (files, paths, alternate indexes, etc.) in a specified user or master VSAM catalog.

DELETE - (VSAM/VTOC DELETE) function allows files to be deleted or reset to empty status using the same selection process used in the 'SAVE/RESTORE=LOGICAL','LISTCAT' and 'LISTVTOC functions.

LVTOC - (LISTVTOC) function provides a compact (but complete) report of the files in the VTOC of one or more disks.

VTMOVE - (VTOC MOVE) function allows the VTOC of a disk to be quickly and easily moved to a new location.

TAPETEST - (TAPETEST) function allows tapes to be tested by writing the chosen blocksize continuously over the length of the reel or cartridge. A report is produced showing the number of 'Erase Gaps' and 'Unrecoverable Errors' encountered for each reel or tape cartridge.

CTAPE - The (TAPECOPY) function allows single or multi-volume tapes to be copied, and if desired, reblocked in the process.

PUNCH - The (Punch) Function provides a quick and easy method for creating IDCAMS control statements for defining VSAM files, and DLBL and EXTENT JCL statements.

LDIR - The (LIST DIRECTORY) produces a library directory report similar to the list directory function of the VSE LIBR program with a number of improvements- a compact 'two up' format is used and0 members may be selected by sub-library, member type, generic member name, and last change date (time stamp).

MCREATE - The mini-disk create function allows the user to create a VSE Mini-Disk under VSE without the use of the IBM DSF utility.

VERIFY - Generates IDCAMS verify for file or groups of files.

BUILD - Invokes the Dr.D VSAM build alternate index feature. This feature will also save, restore, or copy a VSAM cluster during the alternate index build.

SORT - The sort function allows one or more files on a tape or disk backup file (processable by Dr.D) to be extracted and sorted.

DFIX - This function allows the expiration date of selected files to be set to 99/366 so that the files will be considered permanent by the operating system.

`GENERATION=L/F/nnn`

The 'GENERATION=' parameter is used to select disk manager files for backup/restore by the generation number of each file. The allowable values are 'L' for the last (most current) generation, 'F' for the first (oldest) generation, or a number for a specific generation number.

`GR=nnn` `(GLOBAL)`

The 'GR=' (GETVIS Reserve) parameter is used to reserve GETVIS memory so that GETVIS failure will not occur when non-DRD software needs partition GETVIS to perform functions such as tape end of volume or closing of files. The size of the area is specified in thousands of bytes (K)- GR=100 reserves 104,000 bytes of storage. This memory is released before any DRD initiated OPEN, CLOSE, Tape EOV, or end of job.

HIGHKEY= (GLOBAL)

The 'HI=' parameter is used to specify the highest key of a file to restore (see 'LOWKEY=' also). If the key contains spaces, enclose it in quotes ('), if you wish to enter the key in hexadecimal, enter it as HI=X'.....'. Once entered, it applies to the next file restored only. The key should be entered as a contiguous string and be totally contained on one Dr.D control statement.

HV=NO/YES/HOLD (HOLD VTOC)

The 'HV=' parameter is used during volume backup/restore to lock the VTOC of the disk thereby preventing the addition/deletion of files on the disk. This is especially important during compacting of disk manager pool space. HV=NO is assumed. HV=YES should be used on the first disk save/restore within an execution of DR.D and then HV=HOLD should be used to continue to hold the lock on following functions within that DR.D execution.

IDCAMS=PRINT (GLOBAL)

This parameter is used in conjunction with the 'PRINT' keyword to cause the printing of IDCAMS VSAM file definition commands and print output when files are auto defined by Dr.D.

INSEQUENCE=NO

This parameter is used with 'SAVE=VSAM' or 'COPY=VSAM' (FAST=YES must be in effect) to cause KSDS files to be saved/copied without using the 'sequence set' of the file. This is useful for re- covering files whose sequence sets have been corrupted. The user should be aware that this may cause the file to be saved/copied out of sequence, and that for this reason the file will probably need to be sorted and purged of duplicate records (if duplicates exist) before reloading. This can be accomplished by either saving the file to tape (DT=YES specified) or copying to an ESDS file followed by sorting and either copying back from ESDS to KSDS or restoring from tape (DT=YES specified).

LABEL=NO (GLOBAL)

'LABEL=NO' allows the creation of a backup tape without standard labels. This should be used only when a specific need for an unlabeled tape occurs since additional operator responses are required during restore.

LBLOCK SIZE=XXXXX

'LB=XXXXX' where 'XXXXX' is a one to five digit field specifying maximum logical blocksize. This allows variable blocked files to have their logical blocksize adjusted up or down during restore operations. LIBRARY=X

'LIBRARY=X' where 'X' may be 'N' (new),'O' (old),or 'A' (add). For VSE-SP libraries two additional values are allowed- C (create) and 'E' (extend). This entry applies for 'COPY/RESTORE= SC, SR, SP, SS, PC, PR, PS, or LI'. 'N' means create a new library (equivalent to doing a CORGZ)- 'O' means restore to an already existing library on a pack, overlaying its contents- and an 'A' means restore to an already existing library on a pack merging with what's already there. If you add an entry to a library which already has a member with the same name, that

member is deleted and replaced by the new one. 'LIB=N' is invalid for VSE SYSTEM libraries.

The VSE-SP CREATE and EXTEND modes assume that no restoring/ copying are to take place, but that a library is to be Created or Extended (Copy/Restore may follow if desired).

If 'LIB=LIBR' is specified, the restore function will use 'LIBR' facilities, rather than Dr.D physical IOCS routines. (LIB=PH is assumed).

`LOWKEY=` `(GLOBAL)`

The 'LO=' parameter is used to specify the lowest key of a file to restore (see 'HIKEY=' also). If the key contains spaces, enclose it in quotes ('), if you wish to enter the key in hexadecimal, enter it as LO=X'.....'. Once entered, it applies to the next file restored only. The key should be entered as a contiguous string and be totally contained on one Dr.D control statement.

`LRECSIZE=XXXXX`

'LR=XXXXX' where 'XXXXX' is a one to five digit field specifying the logical record length for SD files, so that a logical record count may be produced if used in combination with / / UPSI XX1. Also required to properly restore fixed length 'SD' files to disk (not required for undefined or variable).

`LST=SYSNNN`
`LST=NNN`

The 'LST=' parameter is used to specify a 'SYS' number for listing the Dr.D control statements as entered. It should be entered on an initial 'NOP' statement, and a POWER 'LST' card must be included in the JCL- '* $$ LST CLASS=X,DISP=K,LST=SYS005'.

`MEMBER=X`

'MEMBER=X' where 'X' is either 'S' (Select) or 'R' (Reject). It may be specified whenever saving, copying or restoring a library. If specified, it tells Dr.D to select only those members of the library (Books, Phases, Modules, or Procedures) that meet (Select) or don't meet (Reject) the selection criteria.

Following the control card containing the 'ME=SE/RE' parameter, are the selection control cards having the following formats:

	SOURCE STATEMENT	NON-SOURCE STATEMENT
(S IS SUB-LIBRARY)	123456789	12345678
(MMMMMMMM IS MEMBER)	SMMMMMMMM	MMMMMMMM

An 'S' (Select) or 'R' (Reject) may be placed in column 13 of any control card to override the 'ME=SE/RE' specification, thus allowing subsets of selected groups to be rejected.

`MODIFY=YES`

MODIFY=YES is used to indicate that tape blocks are to be modified during the tape copy function. If specified, one or more pairs of modifier control statements are entered after the FUNCTION=CTAPE, the first of each pair specifying the scan argument and the second specifying the replacement data. The search/replace data can be either character (c'xxxx') or hex (x'xxxx') format.

`MRECORDS=nnnnnnnn` (GLOBAL)

'MRECORDS' is used to specify the maximum number of logical records to be either saved or restored and may be any number in the range 0-99999999. This can be useful for creating test files, or for VSAM, saving or restoring just the definition of a file (MR=0). Once entered, this parameter stays in effect until changed by another 'MR=', I.E. entering 'MR=100' would cause all subsequent files to have no more than 100 records saved/restored/copied. At this writing, this parameter only applies to VSAM files.

`NAME=GEN/NOGEN`

This keyword is used to tell the VSAM file definition function to let IDCAMS generate the data and index names instead of using the data and index names as saved/copied.

`NRECDS=nnn`

'NRECDS=nnn' where 'nnn' is a decimal number specifying a blocking factor. This operand is used to specify the number of logical records per block on disk or tape output.

`NULL=X`

'NULL=X' is used to set the null character used in partial restore control cards. The assumed value is the '/'.

`OPERATOR=XXX...XXX`

'OP=XXX...XXX' where 'XXX...XXX' is a 1 to 80 position message. This optional parameter will log out on SYSLOG. The operator then has the opportunity to 'EOB' and continue processing or hit 'CANCEL' to cancel the job.

This operand may be specified in conjunction with a Save, Copy, or Restore function, in which case it will log out immediately before any processing is done, or it can be on a card all by itself. If the message contains imbedded blanks/commas/equal signs, it must be enclosed in quotes.

`PARTIAL=SELECT/REJECT`

For volume logical Save/Restore, indicates that files are to be selected or rejected by individual and/or generic file ID.

`PID=XXXXXX/NONE`

For volume logical (VTOC) restore, specifies the disk manager 'pool identifier' into which files are to be allocated. Used in conjunction with the 'DM=' (disk

manager) parameter. In some cases, 'PI=NONE' should be used to allow the disk manager to properly process 'cataloged' files.

`PL=CCCTT(CKD)/BBBBBBBBB(FBA)`

For 'CKD PL=CCCTT' where 'CCC' is a three digit decimal number (leading zeroes) representing the number of cylinders to be allocated to a SYSTEM procedure library and 'TT' is a two digit decimal number (leading zero) representing the number of tracks for that SYSTEM library's directory.

For 'FBA PL=BBBBBBDDD' specifies total library blocks and directory blocks for the library. This operand is only required for 'CO/SA/RESTORE=REORG'. It specifies the allocation for the new SYSRES library. If it is omitted, the allocation default is zero and the library is not created. Specifying 'S' (Same) in place of the 'CCCTT/BBBBBBDDD' operand causes Dr.D to use the saved library's allocation.

`PRINT=YES/NO/LIST/BOTH` `(GLOBAL)`

This option allows the user to reverse the assumed/previously specified SYSLST print option. If 'PR=YES/LIST/BOTH' is in effect, a formatted report is produced on SYSLST. If 'PR=LIST' is in effect, non-critical output is directed to SYSLST only. 'PR=BOTH' is used by SP2 library functions to direct member lists to both SYSLST and SYSLOG. In some cases, the print option and upsi settings must be set in advance of the actual Save/Restore operation, so it is best to set the options on a 'NOP' statement immediately after the 'EXEC' statement.

(Note: That UPSI settings are required in conjunction with the print parameter-use UPSI x11xxxxx.)

`R0=YES/NO`

'R0=YES' specifies that the data portion of R0 is to be saved or restored. If the R0 records are in standard format, then it is usually not necessary to restore them, and restore time can be reduced substantially. Certain file types (COBOL direct access and DBOMP) do use the data portion of R0, so care should be used in using this keyword, even though those file types are rarely used, having been replaced by VSAM in most cases.

`RBEGIN=CCCCCCHH(CKD)/BBBBBBBB(FBA)`

The 'CCCCCC' is a zero to six digit decimal number representing a cylinder and 'HH' is a two digit decimal number (leading zero) representing a head. For FBA, the 'BBBBBBBB' is a one to eight digit block number. 'RBEGIN' is required with 'RESTORE/COPY=PHY/ ERASE', specifying a starting cylinder and head number or block at which to begin restoring or erasing. For CKD disk, you may also append 'R' and a 1-3 digit record number in conjunction with 'RMODE=U/A' during COPY/RESTORE=PHYSICAL.

`RCAT=XXXXXXX`

Used during restore of VSAM files using the VSAM catalog logical feature. Specifies the DLBL name (not file ID) of the catalog used to restore into. RESTORE=XX

This keyword is specified when a tape to disk function is required. 'XX' is the operation desired (see "Appendix VIII - Table of Save, Restore, or Copy Commands" on page 207). Only the first two characters of the operand need to be specified.

`RFILENAME=XXXXXXX`

'RF=XXXXXXX' are one to seven alphanumeric characters, the first of which is non-numeric. Whenever it is necessary to specify the name of a file being restored or copied to Dr.D this entry must be included. 'XXXXXXX' is the filename as specified in the DLBL/EXTENT cards. The 'DLBL/EXTENT' cards must be present in either the job stream or the label area. This entry is applicable for 'COPY= or RESTORE=Data', PU, Reorganize, PC, PS, and PR.

NOTE: For copies, 'SFILE= and RFILE=' must be unique or the file copied on top of itself.

`RHALT=CCCCCCHH(CKD)/BBBBBBBB(FBA)`

For CKD disk, 'CCCCCC' is a one to six digit decimal number representing a cylinder and 'HH' is a two digit decimal number (leading zero) representing a head. For FBA Disk, 'BBBBBBBB' is the ending block number. This is required with 'RESTORE/COPY= PHY/ERASE' and denotes an ending track/block at which to terminate restoring or erasing. For CKD disk, you may also append an 'R' and a 1-3 digit record number in conjunction with 'RMODE=U/A' during COPY/RESTORE=PHYSICAL.

`RIDENT=XXX.....XXX`

'RIDENT=XXX...XXX' are 1-44 characters. This specifies the file ID of a library to be copied/restored. This operand may be used in place of 'RFILE=' in certain cases where it might be more advantageous or easier to specify a file-ID instead of using DLBL and extents and specifying the filename. This keyword applies to 'RESTORE= or COPY= PC, PS, PP, or PR'. It is also used to identify the file on the 'RESTORE=EXPIRDATE' function.

`RL=CCCTT(CKD)/BBBBBBDDD(FBA)`

For CKD, 'RL=CCCTT' where 'CCC' is a three digit decimal number (leading zeroes) representing the number of cylinders to be allocated to a SYSTEM relocatable library and 'TT' is a two digit decimal number (leading zero) representing the number of tracks for that SYSTEM libraries directory. For FBA disk, 'BBBBBB' is the total library size in blocks, 'DDD' is the number of directory blocks.

This operand is only specified on 'COPY/RESTORE=REORG' to allocate the new SYSRES library. If it is omitted, the allocation default is zero and the library is not created. Specifying 'S' (Same) in place of the 'CCCTT' or 'BBBBBBDDD' operand causes Dr.D to use the saved libraries allocation.

`RMODE=X (CKD DISK ONLY)`

'RMODE=X' where X is 'I' (Image), 'S' (Squeeze), 'U' (Update) or 'A' (Addon).

This keyword is used to indicate how DR.D is to arrange the data being written on the tracks of the output disk during restore or copy of non-VSAM data files and physical extents. If 'IMAGE' is assumed/specified, the output mirrors the input. If 'SQUEEZE' is assumed/specified, as many blocks as possible are written to each output track. If 'U' is specified, the input block's data is updated into the blocks of output tracks. The 'U' mode is only allowed with RESTORE/COPY=PHYSICAL, and requires the addition of the starting and ending record numbers on the SB/SH and RB/RH parameters. If 'A' is specified, the input blocks are added to the output tracks starting at the record specified by the 'RB' parameter. Squeeze mode is assumed. The 'A' mode is also only allowed with RESTORE/COPY=PHYSICAL, and requires the addition of the starting and ending record numbers on the SB/SH and RB/RH parameters.

This operand is only required with 'COPY/RESTORE=PHY', since it is implied in all other types of saves and restores (i.e. 'RM=I' for volumes, 'RM=I' for Data (DA), and 'RM=S' for data (SD) if Restoring/Copying to a different disk type). Specifying 'RM=I' tells Dr.D to make the restored extent an exact image of the saved one.

Suppose you saved an area from a 3375 and restored it to a 3380. If the 3375 extent had five disk blocks per track, by putting 'RM=I', the 3380 extent will also have five records per track even though it can contain more.

Likewise, if the 3375 had a completely blank track, so will the 3380. In other words, 'RM=I' restores on a track by track basis exactly mirroring the saved extent. On the other hand, specifying 'RM=S' informs Dr.D to use the full track capacity of the restoring device.

In this mode, unused tracks or portions of track on the saved extent are skipped over, and all data records are squeezed on the restoring device, using the full capacity of the track.

While 'RM=' defaults to 'I' unless the file type is SD and the disk type changes, it may be necessary to override that assumption due to a peculiarity in the nature of the file. Therefore, 'RM=' may also be specified on 'COPY/RESTORE=DATA' to override the default.

`RP=YES (Return code Processing)` `(GLOBAL)`

This keyword is used to allow program execution to terminate with a return code even when abnormal termination occurs. This allows conditional JCL to be used in conjunction with the Dr.D return code.

`RSYSNO=SYSXXX`

'RSYSNO=SYSXXX' where 'SYSXXX' is any valid SYSTEM number (see SSYSNO=). This entry is specified when Dr.D needs a 'SYSNO' for an output disk for a Copy or Restore operation- 'ALL, CUR, PC, PS, PR, PP, SC, SS, SR, SP, ER, PH, or ER'.

`RTLOWER=XX`

'RT=XX' where 'XX' is a two digit decimal number (leading zero) representing a lower head limit. It may only be used in conjunction with Copying/Restoring a physical (CO/RE=PHY) when you don't want to restore to an entire extent, but just a split cylinder portion of it.

`RUPPER=XX`

'RU=XX' where 'XX' is a two digit decimal number (leading zero) representing an upper head limit. It may only be used in conjunction with copying/restoring a physical extent (CO/RE=PHY) when you don't want to restore to an entire extent but just a split cylinder portion of it.

```
RVOLID=XXXXXX
RVOLID=(AAAAAA,BBBBBB,CCCCCC,DDDDDD.....)
```

'RV=XXXXXX' where 'XXXXXX' is 6 alphanumeric characters representing a volume serial number. If specified when restoring or copying, Dr.D will locate the disk having the specified volser(s) and make appropriate assignments if possible. If an assignment cannot be made, or the sysno has been specified, and the assumed/specified sysno is assigned to a disk, Dr.D will read the volser and compare it to that specified, requesting that the correct disk be mounted if possible.

For RESTORE=LOGICAL VTOC, up to 30 volsers may be entered to designate the pool of disks to which individual files are to be restored. The pool of disks specified is assumed to be in one to one correspondence with the 'SV=' volsers.

If used during VSAM restore (with auto definition), it allows the user to specify which volumes (of those assigned to the catalog) is to receive the file rather than the normal volume allocation process (see "VSAM Auto Definition Feature" on page 28). If 'RV=NO' is entered, the file definition will exclude any data or index volume specification.

`RWORD=XXXXXXXX`

'XXXXXXXX' are one to eight characters representing the password of the VSAM file being restored to.

`SAVE=XX`

This keyword is specified when a disk to tape function is required. 'XX' is the operation desired. Only the first two (2) characters of the operand need to be specified.

`SBEGIN=CCCCCCHH(CKD)/BBBBBBBB(FBA)`

The 'CCCCCC' is a zero to six digit decimal number representing a cylinder and 'HH' is a two digit decimal number (leading zero) representing a head. For FBA, the 'BBBBBBBB' is a one to eight digit block number. Leading zeros are not required. For CKD disk, you may also append an 'R' and a 1-3 digit record number in conjunction with 'RMODE=UPDATE' during COPY/RESTORE=PHYSICAL.

`SCAT=XXXXXXX`

Specifies the DLBL name (not file ID) of the VSAM catalog to be saved from during 'SA=LO' VSAM backup. During 'RE=LO', it specifies the DLBL having the filed ID of the catalog saved from.

`SEQNO=XX`

'SEQNO=XX' where 'XX' is a two digit decimal number (leading zero). This operand may be specified on any or all Saves, Copies, or Restores. If specified, it checks to see if what is specified equals the sequence number on the output (Save, Copy) or input (Copy, Restore) file. During each save, a sequence number associated with each saved file, pack, library, etc., is incremented by one from its prior save value.

At the initialization of Dr.D, the sequence number is one. During saves and copies, this operand can be used to insure that the control cards are in proper order. During restores, this also insures that the cards are in proper order, matching up with the order of files on the tape.

`SFILENAME=XXXXXXX`

'SF=XXXXXXX' are one to seven alphanumeric characters, the first of which is non-numeric. Whenever it is necessary to specify the name of a file being saved or copied to Dr.D, this entry must be included. 'XXXXXXX' is the filename either as specified in the DLBL/EXTENT cards. The 'DLBL/EXTENT' cards must be present either in the job stream or in the label area. This entry applies to 'SAVE= or COPY=DATA, PC, PS, and PR'.

`SHALT=CCCCCCHH(CKD)/BBBBBBBB(FBA)`

The 'CCCCCC' is a zero to six digit decimal number representing a cylinder and 'HH' is a two digit decimal number (leading zero) representing a head. For FBA, the 'BBBBBBBB' is a one to eight digit block number. Leading zeros are not required. This keyword is used with 'SAVE/COPY=ALL/PHYSICAL' to set the ending disk address of the disk/area being saved/copied. For CKD disk, you may also append an 'R' and a 1-3 digit record number in conjunction with 'RMODE=UPDATE' during COPY/RESTORE=PHYSICAL.

`SIDENT=XXX.....XXX`

'SI=XXX...XXX' are 1-44 characters. This specifies the file ID of a library to be saved/copied. The operand may be used in place of 'SFILE=' in certain cases where it might be more advantageous or easier to specify a file ID instead of using DLBL and extents and specifying the filename. The keyword applies only to 'Save or COPY= PC, PS, PP, or PR', unless used in conjunction with 'FI=' during the restore operation.

NOTE: Any operand containing embedded blanks (most file-ID's do) must be enclosed in quotes:

SI=DOS PRV CORE NO GOOD
SI='DOS PRV CORE' GOOD

`SKIP=XXXXX`

'SKIP=XXXXX' where 'XXXXX' is a three digit decimal number (leading zeros). (Do not use in conjunction with the 'BLOCKSIZE' parameter set the blocksize on a preceding 'NOP'). This operand may be specified for restore functions only. Before restoring DR.D will skip over 'XXXXX' number of files on the tape.

`SLIBRARY=CCCTT(CKD)/BBBBBBDDD(FBA)`

For 'CKD-SL=CCCTT' where 'CCC' is a three digit decimal number (leading zeroes) representing the number of cylinders to be allocated to SYSTEM source statement library and 'TT' is a two digit decimal number (leading zero) representing the number of tracks for that SYSTEM libraries directory.

For 'FBA Disk- BBBBBB' is the total library size in blocks, 'DDD' the directory size in blocks. This is only specified on 'CO/RE=REORG'. It specifies the allocation for the new SYSRES library. If it is omitted, the allocation default is zero and the library is not created.

Specifying 'S' (Same) in place of the 'CCCTT/BBBBBBDDD' operand causes Dr.D to use the saved libraries allocation.

`SMODE=XXX`

The 'SM=' (SELECT MODE) keyword is used to change the selection logic when files or library members are selected for processing. At this time, only the 'VTOC driven' functions use this keyword, with 'SMODE=A' used to cause files to be selected only if they begin on one of the specified volumes.

`SOPTION=DATA/INDEX  (SHARE OPTION)  (GLOBAL)`

The 'SOPTION=DD/II' keyword is used to override the VSAM share options used to define files by the Dr.D auto define feature. The data share options are entered first, and the index next, with the '/' separating them. This is a global option and remains in effect throughout the execution or until reentered (use SO=N to terminate its use).

`SPACE=n`

This parameter is used to control spacing on the LISTCAT and the LISTVTOC reports. The report normally prints one blank line in between files, so enter a value from 2-9 to cause additional blank lines between files.

`SPLIT=NO`

Used during NON-VSAM RESTORE=LOGICAL to prevent 'SD' files from being split into multiple extents when a reorganizing restore is performed. ('DA' files are never split).

`SSYSNO=SYSXXX`

'SSYSNO=SYSXXX' where 'SYSXXX' is any valid SYSTEM number, (SYSRDR, SYSIPT, SYSPCH, SYSLST, SYSLOG, SYSRES, SYSRLB, SYSSLB, SYSUSE, SYSREC, SYSLNK, SYSCLB, SYSVIS, SYSCAT, or SYS000, SYS001, etc.). This

entry is specified when Dr.D needs a 'SYSNO' for an input disk for a save or copy operation.

`STLOWER=XX`

'ST=XX' where 'XX' is a two digit decimal number (leading zero) representing a lower head limit. It may only be used in conjunction with saving/copying a physical extent (SA/CO=PHY) when you don't want to save the entire extent, but just a split cylinder portion of it.

`SUPPER=XX`

'SU=XX' where 'XX' is a two digit decimal number (leading zero) representing an upper head limit. It may only be used in conjunction with Saving/Copying a physical extent (SA/CO=PHY) when you don't want to save the entire extent but just a split cylinder portion of it.

```
SVOLID=XXXXXX
SVOLID=(AAAAAA,BBBBBB,CCCCCC,DDDDDD.....)
```

'SV=XXXXXX' where 'XXXXXX' is 6 alphanumeric characters representing a volume serial number. If specified when saving or copying, Dr.D will locate the disk having the specified volser(s) and make appropriate assignments, if possible. If an assignment cannot be made, or the sysno has been specified, and the assumed/specified sysno is assigned to a disk, Dr.D will read the volser and compare it to that specified, requesting that the correct disk be mounted if possible.

For save logical VTOC, up to 30 volsers may be entered to designate the pool of disks from which individual files are to be saved.

`SWORD=XXXXXXXX`

'XXXXXXXX' are one to eight characters representing the password of the VSAM file being saved.

`SYNC=YES`

'SYNC=YES' ('YES' is the only valid operand) is specified when 'COPY= or RESTORE=DATA' (SD and DA files). Normally Dr.D will treat a multi-extent output disk area as one contiguous prime data area. Specifying 'SYNC=YES' will cause the same number of records to be written in each extent as were read from the saved extent (the save and restore extents are synchronized). Thus, when 'SYNC=YES' parameter is used the number of extents in the saved and restored files must be the same and each restore extent must be large enough to contain the number of records to contain its corresponding save extent.

`TAPE=XXX`

'TAPE=XXX' where 'XXX is either REW, RUN, or NOR'. Not valid if 'TAPEIN or TAPEOUT' are disk files.

'TAPE=XXX' when used in conjunction with 'FUNCTION=TAPETEST', 'XXX' is the number of tapes to be tested at any given execution of the Tapetest function.

It may be specified for either Saves or Restores and is ignored for copies. If specified, the input (Restore) or output (Save) tapes are closed. Normally Dr.D will open with 'NORWD' when called for, and close with 'NORWD' at termination for both input and output tapes.

This operand allows you to Read/Write multiple DOS tape files for one execution of Dr.D. 'XXX' represents the disposition of the tape at close time.

`TDDNAME (tape DD or DTF symbolic name)` (GLOBAL)

'TDDNAME=DDDDDDD' where 'DDDDDDD' is the desired name (rather than 'TAPEIN/TAPEOUT' or the Data Tape generated name). If the specified function is 'SAVE' then the TAPEOUT name is replaced. If restoring, then the TAPEIN name is replaced. In either case, if the name being replaced is open, it is closed prior to replacement and subsequent open of the tape file.

Multiple DD names may be entered by separating them with the '/' character. All three names can be specified, for instance, TD=TAPEOUT/TAPETWO/TAPE003, or any combination as long as the '/' is present- TD=//TAPEXXX for instance would specify the name for the third tape while TD=/TAPEYYY would specify the name for the second tape. If the TD= keyword is used to specify the name for either TAPETWO or TAPE003, then the UPSI setting is not needed to invoke the creation of the additional tape(s). Invoking the creation of the third tape always implies creation of the second tape, i.e. specifying UPSI=XXXXXXXX1 is equivalent UPSI=XXX1XXXX1 and specifying TD=//TAPE003 is the same as TD=/TAPETWO/TAPE003.

`TI/TO/TT/T3=SYSXXX (TAPEIN/TAPEOUT/TAPETWO/TAPE003)` (GLOBAL)

To specify a tape SYSNO different from that assumed. This parameter must be used on the first control card or repeated on the 2nd if the 1st card is 'SAVE=IPL'. It is then effective throughout an execution of Dr.D.

`TO=cccchh (CKD disk only)`

For RESTORE=ALL/CURRENT with AREA=SELECT, this allows the selected area to be restored to a new cylinder and head location. If the 'cccchh' is preceded by the letter 'K', the cylinder number of the SAVED area is retained. This parameter is added to the AREA selector card, not the RESTORE=ALL/CURRENT statement.

`TSTAMP=XXXXXXXX (TIME STAMP)` (GLOBAL)

The 'TSTAMP=XXXXXX' keyword is used to select VSAM files or library members based on their time stamps. At this time, it applies to:

SAVE/RESTORE=LOGICAL,
FUNCTION=LCAT,
FUNCTION=DELETE,FUNCTION=LDIR,
SAVE/RESTORE=LIB.

The operand is either an absolute date or a date relative to today (the machine IPL date). If an absolute date, the format is YYDDD where YY is the year and

DDD is the Julian day of year. If the date is relative, the format is -DDD and DDD is the number of days relative to the IPL date. In either case, the entered date may be suffixed with '.HHMM' where HH is the hour of the day (00-23) and MM is the minute of the hour (00-59). If the '.HHMM' is not entered, the hour and minute are set to '0000'. If the date is prefixed by '>' (greater than) then the member/file is selected only if its time stamp is greater than that entered. If the date is prefixed by '<' (less than) then the member/file is selected only if its time stamp is less than that entered. If neither '<' or '>' is used, then the member/file is selected only if its time stamp year and day are equal to that entered, and the hour and minute are not used for selection. Examples:

TS=-0	select if stamped today
TS=>-1.1200	select if stamped after yesterday at 12 noon
TS=>-6	select if stamped in the last seven days
TS=91100	select if stamped on the 100th day of 1991
TS=>91100.1450	select if stamped after 2:50 pm on 91/100
TS=<91100	select if stamped before 91/100

In order to fully understand this keyword, it is suggested that you first use it with the FUNCTION=LDIRECTORY and FUNCTION=LCAT, since both reports list the time stamp of the file/member.

`UPSI=NNNNNNNN` (GLOBAL)

For Dr.D, this allows the UPSI to be specified on the control card rather than in the job stream.

`UNIQUE=YES/NO/A`

UNIQUE=YES is specified with the BUILDAIX function to indicate that a four byte binary sequential number is to be appended to each AIX key generated to insure that each key is unique. UNIQUE=NO is the assumed value. UNIQUE=A is used to specify that duplicate keys are to be dropped if the AIX is defined as UNIQUE. This keyword may be added to the FUNCTION=BUILD control statement, or if PARTIAL=SELECT is specified, for each AIX of a multiple build.

`USER=PHASENAM/MM/TZZZZZZ`

'USER=PHASENAM/MM/TZZZZZZ' is specified when user-exits are to be invoked and the user-exit special feature is installed. Parameter values are:

'PHASENAM'	is the cataloged name of the user-program.
'MM'	is the mode- 'IO' to send/receive logical records 'EX' to inspect logical records.
'T'	is the core image type catalog- 'L' for relocatable or self relocating programs- 'A' for absolute address programs.
'ZZZZZZ'	is program size- decimal length for type 'L' programs- hexadecimal load address for type 'A' programs.

`VARIABLE=YES`

When saving VSAM files, if the records of the file are variable, 'VA=YES' must be specified.

`VBUFF=NNN` `(GLOBAL)`

For VSAM restore operations, this allows the buffer space to be increased to enhance performance. The NNN is a one to three digit number specifying the size in thousands of bytes-'VB=100' sets the buffer to 100,000. If not specified, the buffer space is set to three times the assumed or specified tape blocksize (see 'BL' parameter). For VSAM save the 'FAST=YES' function normally uses a buffer equal to the tape blocksize, but the 'VB=' parameter can be used to increase the buffer size with possible enhanced performance.

`VCOMPRESS=YES`

When restoring VSAM files to VSE-ESA with the VSAM compression feature, DR.D will specify the compression attribute for any file it defines/redefines if the file qualifies for compression. Dr.D will determine whether the file qualifies by using the rules specified in VSAM documentation. This allows files to be restored or copied into compressed format without the need to pre-qualify those files.

The 'FD=' keyword should be used to insure that file re-definition is performed.

`VNULL=YES/NO/ALL`

Specifies action to be taken when null VSAM files are encountered. 'YES' (the assumed option) indicates that null files are to be considered for selection. 'NO' indicates that null files are not to be considered for selection. If used in conjunction with selection by volume (SV=xxxxxx), VNULL=ALL causes null files to be treated as though they are selected by volume.

`VOLSER=XXXXXX`

'VOLSER=XXXXXX' is used to set the serial number of the disk being restored or copied to (using 'SA=ALL' or 'SA=CUR'), to a different value than that Saved/Copied.

```
VOLSELECT=SSSSSS
VOLSELECT=(VOLSR1,VOLSR2,...)
```

This form of the 'VO' parameter is used with SAVE LOGICAL VTOC to select files that fully or partially reside on the specified disk(s) during backup, and likewise to select the files for restore based on the disk(s) they resided on during backup.

`VRTYPE=K/E/R/S`

'VR=' specifies the VSAM file type for restoring VSAM files as follows:

K Keyed files (the assumed value)
E Entry sequenced files
N Non-CI format files
R Relative record files

S VSAM managed sequential
P PATH

VSAM=YES

If specified during 'SAVE=LOGICAL (VTOC)' processing, VSAM data areas are saved as 'DA' files. (VSAM=NO is assumed). This method of saving VSAM files is not recommended as the VSAM area is saved as a DA file and must be restored as DA. Use 'SAVE=LOGICAL VSAM' instead. VSAM=NO

If specified during 'SAVE=CURRENT' processing (CKD disk only), the VSAM space is not saved ('VSAM=YES' is assumed).

VSAM=SET

If specified with 'RESTORE=EXPIRATIONDATE', the format one label for the specified file is changed to designate VSAM file type.

VSTYPE=K/E/R/S

'VS=' specifies the VSAM file type for saving VSAM files are as follows:

K Keyed files (the assumed value)
E Entry sequenced files
N Non-CI format files
R Relative record files
S VSAM managed sequential
P PATH

VUSE=CA/IG/NU/SA/OP

'VU=' specifies the option to be taken if VSAM files being saved are not shareable.

The assumed value is 'CA' (Cancel) Dr.D will cancel with File Id in use. 'IG' (Ignore) will cause any file in use to be ignored. 'NU' (NULL) Dr.D will save the object as a null file without issuing any message. If 'FAST=YES' is specified, then two additional options are available 'SA' (SAVE) causes Dr.D to attempt to save the file even though it may be in use by another task, while 'OP' (Operator) allows the operator to specify the action to be taken. If 'VUSE=SAVE' is specified and the file is being updated by another task, results are unpredictable and may cause the backup to fail.

WAIT=NNNN (GLOBAL)

In order to lessen interference with heavily I/O bound teleprocessing systems, the 'WAIT' parameter allows the user to specify a pause between each tape I/O, so that teleprocessing disk I/O is not locked out. Wait time is specified in 300ths of a second. Thus 'WAIT=0300' would cause a one second delay after each tape I/O.

WORK=XXXXXX (GLOBAL)

This keyword is used to specify the number of sort work files and their starting SYSNO's for the VSAM AIX build function. The first one or two digits specify the

number of work files to be used. If the starting SYSNO is specified, a '/' is coded next, followed by a one to three digit number in the range of 1-253. Example: WORK=2/1 indicates that two sort work files are used with SYS001 and SYS002 assigned respectively.

`WR=PPPPPPPP` (GLOBAL)

This keyword is used to enter the VSAM catalog password for the catalog being restored to or deleted from.

`WS=PPPPPPPP` (GLOBAL)

This keyword is used to enter the VSAM catalog password for the catalog being saved or listed from.

`XL=YES/NO`

For VSAM restore, this keyword specifies whether KSDS files are to be defined with the 'EXTRALARGEDATASET' attribute. If 'NO' is specified, then the attribute is suppressed even if it was present in the saved file. This keyword is effective only if an IDCAMS define is initiated- use the 'FD=' keyword to force delete/define in case the file exists in the receiving catalog.

`XS=N`

For RESTORE=LOGICAL (VTOC), this keyword allows the starting extent sequence number for each file to start at other than the assumed value zero.

Appendix II - Setting Assumed Options

Dr.D assumed options can be customized by the user through the DRZAP phase modifier utility. These options are stored in the phase DRD and in the past were customized by DRZAP or other utilities such as the IBM MSHP program. DRZAP can now do this customization in a more 'user friendly'/interactive fashion, either under CICS or in batch mode. If a corresponding DR.D keyword exists for any option, the value entered will be the same as those specified in the keyword section. As each existing option is displayed, enter the desired change or nothing to proceed to the next option or enter an asterisk (*) to backup to the preceding option.

It is usually prudent to make a copy of phase DR.D in a test sub-library before changing any options, in case any errors are made that could cause problems with jobs that might execute while the options are being changed.

Once DRZAP begins execution, enter the library, sublibrary and 'DRD'. Example- 'PRD2.DRDTST.DRD'. Then enter the command 'OPT'. DRZAP will respond as follows:

.....DRZAP DISPLAY.....	**....ALLOWABLE ENTRY....**
OLD PRINT OPTION: NO , ENTER NEW	**NO YES LIST**
This option changes the assumed value for keyword 'PRINT'	
OLD UPSI SETTINGS: 00000000 ,ENTER NEW	**00000000-11111111**
Changes the assumed value for keyword 'UPSI''	
OLD UPSI MASK: 11111111 ,ENTER NEW	**00000000-11111111**
Specifies the UPSI setting to be used from the JCL // UPSI statement. I.E. if 11111111 is set, all JCL UPSI switches will be used, but if 00000000 is set, no JCL UPSI switches will be used.	
OLD TAPE BLOCKSIZE: 065496 ,ENTER NEW	**016000-065496**
Changes the assumed value for keyword 'BL'.	
OLD REWIND OPTION: NOREWIND ,ENTER NEW	**REWIND NOREWIND RUN**
Changes the assumed tape rewind option (NOREWIND)	
OLD TAPEIN SYS NUMBER: SYS008 ,ENTER NEW	**SYS000-SYSnnn**
Changes the assumed value for keyword 'TI'. (SYSnnn is the user system limit.)	
OLD TAPEIN DDNAME: TAPEIN ,ENTER NEW	**1-7 CHARACTER STRING**
Changes the assumed tape 'DD' name.	
OLD TAPEOUT SYS NUMBER: SYS007 ,ENTER NEW	**SYS000-SYSnnn**
Changes the assumed value for keyword 'TO'. (SYSnnn is the user system limit.)	
OLD TAPEOUT DDNAME: TAPEOUT ,ENTER NEW	**1-7 CHARACTER STRING**
Changes the assumed tape 'DD' name.	

.....DRZAP DISPLAY.....	....ALLOWABLE ENTRY....
OLD TAPETWO SYS NUMBER: SYS006 ,ENTER NEW	**SYS000-SYSnnn**
Changes the assumed value for keyword 'TT'. (SYSnnn is the user system limit.)	
OLD TAPETWO DDNAME: TAPETWO ,ENTER NEW	**1-7 CHARACTER STRING**
Changes the assumed tape 'DD' name.	
OLD TAPE003 SYS NUMBER: SYS005 ,ENTER NEW	**SYS000-SYSnnn**
Changes the assumed value for keyword 'T3'. (SYSnnn is the user system limit.)	
OLD TAPE003 DDNAME: TAPE003 ,ENTER NEW	**1-7 CHARACTER STRING**
Changes the assumed tape 'DD' name.	
OLD FAST OPTION: YES ,ENTER NEW	**YES NO**
Changes the assumed value for keyword 'FAST'.	
OLD COMPRESS OPTION: NO ,ENTER NEW	**YES NO**
Changes the assumed value for keyword 'CP'.	
OLD VSAM FILE DELETE FUNCTION: RESET ,ENTER NEW	**RESET DELETE CONSOLIDATE**
Changes the assumed value for keyword 'FD'.	
OLD AUTO ASSIGNMENT OPTION: YES ,ENTER NEW	**NO YES**
This option should be 'NO' if a disk manager system is used.	
OLD VARIABLE OPTION: NO ,ENTER NEW	**NO YES**
Changes the assumed value for keyword 'VA'.	
OLD SYS001 SYS NUMBER: SYS001 ,ENTER NEW	**SYS000-SYSnnn**
Changes the assumed value for keyword 'SS'. (SYSnnn is the user system limit.)	
OLD SYS002 SYS NUMBER: SYS002 ,ENTER NEW	**SYS000-SYSnnn**
Changes the assumed value for keyword 'RS'. (SYSnnn is the user system limit.)	
OLD VUSE OPTION: CANCEL ,ENTER NEW	**CANCEL OPERATOR IGNORE SAVE NULL**
Changes the assumed value for keyword 'VU'.	
OLD CONTROL TABLE SIZE: 004088 ,ENTER NEW	**4088-65496**
Changes the size of the table used to store control statements when UPSI xxxxxx1 is set. Allows more statements to be entered. The execute size must increase by the increase in size.	
OLD REPORT SYS NUMBER: SYSLST ,ENTER NEW	**SYS000-SYSnnn or SYSLST**
Changes the assumed 'SYS' number for print output. (SYSnnn is the user system limit.)	
OLD DOCTOR 126 PRINT OPTION: YES ,ENTER NEW	**YES NO**
Determines whether the DOCTOR 126 message prints.	
OLD DOCTOR 149 PRINT OPTION: YES ,ENTER NEW	**YES NO**
Determines whether the DOCTOR 149 message prints.	

.....DRZAP DISPLAY.....	....ALLOWABLE ENTRY....
OLD DRD POOL OPTION: NO ,ENTER NEW	YES NO
Determines whether the DRD DISK POOL feature is to be invoked for non-VSAM file functions.	
OLD ASSOCIATIONS OPTION: NO ,ENTER NEW	YES NO
Changes the assumed value for keyword 'AS'.	
OLD GETVIS RESERVE SIZE (K): 00000 , ENTER NEW	000-999
Changes the assumed value for keyword 'GR'.	
OLD RETURN CODE OPTION: ,ENTER NEW	YES NO
Changes the assumed value for keyword 'RP'.	
OLD CTL LST SYS NUMBER: NONE ,ENTER NEW	SYS000-SYSnnn
Changes the assumed value for keyword 'LS'. (SYSnnn is the user system limit.)	

At this point you may restart the process by entering 'R' or hit enter to end the session.

You may also at any time during the session enter 'END' to terminate the session and save the changes made, or you may enter 'QUIT' to terminate without saving the changes.

Appendix III - Dr.D POOLS Feature

For non-VSAM file functions, the DRDPOOLS feature allows up to ten pool identifiers, each with up to thirty disks assigned to it. Once established, and the DRD pool option is set, DRD will convert any pool id, entered via the SVOLSER or RVOLSER keywords, to the disk volsers defined by the user.

The pool-ids and their associated disk volsers are entered via the DRZAP utility in much the same way that the assumed options are set, described earlier in this section. The entered data is stored in a PHASE called DRDPOOLS, provided with DRD.

To update the pool information in the DRDPOOLS PHASE, invoke DRZAP and enter the usual library.sublibrary.phase information, for example PRD2.DRDV643.DRDPOOLS. Then enter the command "POOL". DRZAP will then display the existing pool and ask for a pool update command. The "ADD" command is used to initially define a pool id, and to add additional disk volsers later. The "DELETE" command is used to fully delete a pool by entering "DELETE POOLID,ALL" or to delete individual disk volsers by entering "DELETE POOLID,VOLSR1,VOLSR2,....". No more than seven volsers may be added or deleted in each command, and the pool id and volsers must each be six characters long. The command "DELETE" may also be entered as "DEL".

To allow use of the entered pools, you must set the DRD POOLS option as specified in the preceding page(s).

Appendix IV - Dr.D Operating Environment

Dr.D is compatible with any IBM system ES90xx, 30xx, 43xx, or plug compatible CPU operating under the VSE-ESA, VSE-SP, VSE, or DOS-MVT-VSE operating systems, and any disk or tape device supported by those operating systems.

Dr.D will operate with or without the following system options:

a. Seek separation

b. DASD file protect

c. Asynchronous processing (AP=YES - FOR MULTITASKING)

d. RPS support (VS)

e. PFIX support (VS)

f. Spooling systems (POWER, GRASP, SPRINT, etc.)

Dr.D may be invoked in any partition such as:

a. From the console (SYSLOG) where it will begin reading control cards from SYSLOG.

b. From the card reader (SYSRDR) where it will begin reading control cards from SYSIPT.

c. Or from the PROC library where it will begin reading control cards from the procedure if present, else from SYSIPT.

Appendix V - Setting UPSI Bits

Dr.D tests all eight UPSI bits to determine if any special action is required.

If bit 0 (/ / UPSI 1) is on, Dr.D does not wait for an operator reply after messages 16 or 66.

If bit 1 (/ / UPSI x1) is on, Dr.D will log all the control cards it reads onto SYSLOG (and SYSLST if the print option is set).

If bit 2 (/ / UPSI xx1) is on, Dr.D will log out statistics about the save, restore, or copy at its conclusion, on SYSLOG (AND SYSLST if desired).

If bit 3 (/ / UPSI xxx1) is on during a save, a duplicate backup tape will be created on SYS006, / / TLBL TAPETWO.

If bit 4 (/ / UPSI xxxx1) is on during volume SAVE/COPY operations, data checks during read data operations are noted on the console log. The operator is then given the option to RETRY, CANCEL, or IGNORE the error.

If bit 5 (/ / UPSI xxxxx1) is on during volume SAVE/COPY operations, data checks during read data operations are noted on the console log. The error is automatically ignored but is noted on the console.

NOTE: Ignoring data checks may produce unpredictable results and is not generally recommended. Data integrity is usually lost, and files having pointers may be unusable.

If bit 6 (/ / UPSI xxxxxx1) is on during file restore operations, a table of control cards is built allowing restore control cards matched against files on the input save tape using either the 'RF=' name or 'FIND=' name. **Note: This UPSI should be set in the job control statement.**

Only individual file restores are allowed, and are limited in number to those that will fit in the control card table (See "Appendix II - Setting Assumed Options" on page 199). To determine the maximum number, use the sum of the truncated card length plus one.

Bit 7 (/ / UPSI xxxxxxx1) is used to override the 'PRINT=LIST' option that directs all printed output to SYSLST with no SYSLOG output. If on, control cards are printed on SYSLOG even though 'PRINT=LIST' is specified.

Dr.D allows the UPSI bits to be specified on the control card as well as the JCL. The same character specification 1, 0 and X apply. A default UPSI specification is included with the utility and can be changed at installation time.

Appendix VI - DOS/VSE OPERATIONS

The following are some considerations for operating Dr.D under DOS/VSE/ESA:

A. Dr.D may be executed in real - / / EXEC DRD,REAL - or virtual storage - / / EXEC DRD

B. Dr.D should be executed with size=nnnK where 'nnn' is at least 200, and should be increased by 2 for each 1,000 byte increase in tape blocksize above the assumed 65,496 byte tape blocksize. Increasing the EXEC size beyond 200K is of no benefit and may degrade performance due to reduction in partition GETVIS.

C. If Dr.D is executing in the virtual mode, and 'PFIX' support is generated into the supervisor, it will attempt to fix as much as possible of the I/O buffers and frequently used code in real storage.

The Dr.D system consists of the following phases:

DRD	DMANPRNT	DOCTOR26	DOCTOR56*
DRVIS	DOCTOR00	DOCTOR27**	DOCTOR57*
DRZAP	DOCTOR01**	DOCTOR29*	DOCTOR58
DRZAPC	DOCTOR02**	DOCTOR31*	DOCTOR59
DUMPSERV	DOCTOR03**	DOCTOR33*	DOCTOR60
$$BGSI01	DOCTOR04**	DOCTOR34	DOCTOR61
$$BGSI02	DOCTOR05	DOCTOR35	DOCTOR62
$$BGSI03	DOCTOR06**	DOCTOR36*	DOCTOR63
$$BGSI07	DOCTOR07**	DOCTOR37*	DOCTOR64
$$BGSI08	DOCTOR08	DOCTOR38*	DOCTOR67
$$BGSI09	DOCTOR09**	DOCTOR39*	DOCTOR68
$$BGSI10	DOCTOR10	DOCTOR40*	DOCTOR69
$$BGSI11	DOCTOR11**	DOCTOR41*	DOCTOR70
$$BGSI12	DOCTOR12**	DOCTOR42*	DOCTOR71
$$BGSI13	DOCTOR13	DOCTOR43*	DOCTOR72
$$BGSI14	DOCTOR14	DOCTOR44*	DOCTOR73
$$BGSI15	DOCTOR15	DOCTOR45*	DOCTOR74
$$BGSI16	DOCTOR16	DOCTOR46**	DOCTOR75
$$BGSI17	DOCTOR17	DOCTOR47	DOCTOR76
$$BGSI18	DOCTOR18**	DOCTOR48	DOCTOR97
$$BGSI19	DOCTOR19**	DOCTOR49	DOCTOR98
$$BGSI20	DOCTOR20**	DOCTOR50	DOCTORCN
$$BGSIAV	DOCTOR21**	DOCTOR51	DOCTORMS
$$BGSIVS	DOCTOR22	DOCTOR52**	DOCTORPC
$OCTOR95	DOCTOR23	DOCTOR53**	DOCTORSF
$OCTOR96	DOCTOR24	DOCTOR54**	DOCTORVL
$OCTOR99	DOCTOR25	DOCTOR55*	DOCTORWW

** CKD ONLY PHASES
* FBA ONLY PHASES

Appendix VII - Control Card Format

Dr.D performs one operation per 'logical' card. Any number of 'logical' cards may be submitted on one execution of Dr.D, If an operand contains embedded blanks it must be contained within quotes. The keyword pairs on a card do not have to be in any particular order.

Dr.D reads 80-position control statements from either SYSIPT or SYSLOG. If the '/ / EXEC DRD' is read from SYSRDR then Dr.D starts reading from SYSIPT. If the '/ / EXEC DRD' was keyed in on SYSLOG, it reads from SYSLOG.

To switch reading from SYSIPT to SYSLOG (or vice versa), insert a control statement with '$$$$$$' in the first six positions of the control card. A '/*' from either SYSIPT or SYSLOG terminates execution.

All other control statements have the following format which always begins in column one (CC1).

```
Col--> 1........10........20........30........40........50....
Data-> KEYWORD1=OP1,KEYWORD2=OP2,KEYWORD3=OP3,...,KEYWORDX=OPX
```

Each keyword is followed by '=' followed by its operand. A comma must follow each keyword pair except the last which must be followed by a blank.

Continuation Cards

Cards may be continued by specifying an asterisk (*) as the last keyword (i.e. after the last comma). This step may be repeated as many times as necessary.

Abbreviations

Since Dr.D only looks at the first two positions of each keyword, the statement:

```
SAVE=CURRENT,SSYSNO=SYS005,EXPIRED=YES,OPERATOR=HELLO
```

and

```
SA=CU,SS=SYS005,EX=YES,OP=HELLO
```

are equivalent.

Appendix VIII - Table of Save, Restore, or Copy Commands

Save (Only) Command

SAVE=	STANDING FOR:	MEANING:
IP	IPL Tape	Writes IPL records on front of tape for subsequent standalone restore.

Combination of Save, Restore, or Copy Commands

SA/RE/COPY=	STANDING FOR:	MEANING:
AL	ALL	SA/RE/COPY an entire disk.
CU	CUrrent	SA/RE/COPY active portions of a disk plus VTOC and IPL track.
DA	DAta	SA/RE/COPY a DATA file utilizes 'PIOCS' for I/O.
SD	SDisk	SA/RE/COPY a SEQUENTIAL DISK file utilizes 'LIOCS' for I/O.
VS	VSam	SA/RE/COPY a VSAM file.
IS	ISam	SA/RE/COPY a ISAM file using logical IOCS.
LO	LOgical	(SA/RE ONLY) VSAM catalog or VTOC logical backup.
PH	PHysical	SA/RE/COPY a PHYSICAL area of a disk.
NO	NO-operation	SA/RE/COPY for entry of various tape and GLOBAL functions.
LI	VSE-SP LIbrary	SA/RE/COPY VSE-SP library.
RE	REorganize	SA/RE/COPY IPL track, SYSTEM directories and label area. (VSE. 1.3.5 ONLY)

(VSE 1.3.5 Only) Commands

SA/RE/COPY=	STANDING FOR:	MEANING:
SC	System Core	SA/RE/COPY CORE library.
SR	System Relo	SA/RE/COPY RELO library.
SS	System Source	SA/RE/COPY SOURCE library.
SP	System Proc	SA/RE/COPY PROC library.
PC	Private Core	SA/RE/COPY CORE library.
PR	Private Relo	SA/RE/COPY RELO library.
PS	Private Source	SA/RE/COPY SOURCE library.
PP	Private Proc	SA/RE/COPY PROC library.

Restore (only) Commands

RESTORE=	STANDING FOR:	MEANING:
PU	PUt format-1	CREATES a format-1 label for DA or SD files.
ER	ERase	CLEARS areas of a disk.
EX	EXpiration date	ALTERS expiration date of a file.

(Function= Commands)

FUNCTION=	STANDING FOR:	MEANING:
LCAT	LISTCAT	Compact listing of (files, paths, alternate indexes, etc...)
DE	DELETE	Allows files to be deleted or reset to empty status.
LVTOC	LISTVTOC	Provides a compact (but complete) report of the files in the VTOC of one or more disks.
VTMOVE	VTOCMOVE	Allows the VTOC of a disk to be quickly and easily moved to a new location.
TAPETEST	TAPETEST	Allows tapes to be tested by writing the chosen blocksize continuously over the entire length of the reel or cartridge. The report shows the number of erase gaps and unrecoverable errors encountered on each cartridge or reel.
CTAPE	TAPECOPY	Allows single or multi-volume tapes to be copied, and if desired, reblocked in the process.
LTAPE	LISTTAPE	Lists the contents of a backup tape.
PUNCH	PUNCH	Provides a quick and easy method for creating IDCAMS control statements for defining VSAM files, and DLBL for EXTENT JCL statements.
LDIRECTORY	LIST DIRECTORY	Provides a library directory listing similar to the LIBR directory listing but with a number of improvements such as a 'two up' format. Members may be selected by sub-library, member type, generic member name, and last change date (time stamp).
VERIFY	VERIFY	Generates IDCAMS verify for file

		or group of files.
BUILD	BUILDAIX	Invokes the Dr.D VSAM build alternate index feature. This feature will also save, restore, or copy a VSAM cluster during the alternate index build.
SORT	SORT	The DR.D sort function allows one or more files on a tape or disk backup file (processable by DR.D) to be extracted and sorted.
MC	MCREATE	The mini-disk create function allows the user to create a VSE Mini-Disk under VSE without the use of the IBM DSF utility.
DFIX	DATEFIX	This function allows the expiration date of selected files to be set to 99/366 so that the files will be considered permanent by the operating system.

NOTE: 1 Specifying 'RE=AL' or 'RE=CU' are equivalent. Whether the disk is restored in its entirety or just the active portions depends on whether 'SA=AL' or 'SA=CU' was in effect when it was saved.

NOTE: 2

```
SAVE=ERASE                    SAVE=EXPIRDATE
COPY=ERASE                    COPY=EXPIRDATE
SAVE=CCF                      RESTORE=IPLTAPE
COPY=CCF                      COPY=IPLTAPE
SAVE=PUT FORMAT-1 LABEL       COPY=LOGICAL
COPY=PUT FORMAT-1 LABEL
```

ARE ALL UNDEFINED AND INVALID

NOTE: 3 Only the first two (2) characters of the keyword and operand are examined -

SA=IP or SAVE=IPLTAPE

SA12345=IPXYZ are equivalent and acceptable.

Appendix IX - Dr.D Conversion Utility

This Appendix describes how to use the Dr.D conversion utility that will convert OEM Dump/Restore job control to Dr.D format. Westinghouse Disk Utility System is a registered trademark of Westinghouse, Inc. Maxback, DUSP are registered trademarks of Blueline Software. Faver is a registered trademark of Computer Associates Inc.

A set of programs have been developed that will allow users of various OEM Backup/Restore utility packages to implement Dr.D without manually converting their existing Backup/Restore job streams. The Dr.D conversion utility automatically converts job streams, of various OEM Backup/Restore products, to the Dr.D format. The Dr.D conversion utility will allow users of other OEM Backup/Restore package to implement Dr.D immediately, eliminating the need to convert existing Backup/Restore jobs.

How The Conversion Utility Works

To convert existing jobs to Dr.D format, the user simply executes the DR.D conversion utility with the existing jobs as input. Each job supplied in the input must have a /* delimiter. An end-of-data (EOD) statement is used to indicate end of input file to DRDCNVCU. The DR.D conversion utility will read the input, convert it to Dr.D format, and punch a new set of JCL. The DR.D conversion utility will not read the existing JCL from any source other than SYSIPT. Several examples of how to create the conversion job are included with this documentation. A report listing the input statements processed and the diagnostic errors generated is produced during the conversion run.

Using The DR.D Conversion Utility

The batch program 'DRDCNVCU' is executed to perform the actual conversion. The jobs to be converted are placed after the // EXEC statement along with a conversion control statement. DRDCNVCU reads all of the input statements, converts the utility control statements to Dr.D format, modifies the // EXEC statements for Dr.D execution, and outputs the new job streams to the punch queue. Power JECL statements must be changed from '* $$' to '* **' if they are to be included in the conversion job stream. SYS004 must be assigned to the reader and SYSPCH must be assigned to the punch.

Conversion Control Statement

This must be the first control statement in the conversion job stream unless specified via a '/ / EXEC statement parm'.

```
CONVERT=aaaaaa,EOD=bbb,EOJ=NO/YES,PWREOJ=NO/YES,CMS=ccc,SAVE=,TFNI=,TFNO=
```

`COVERT=aaaaaa` specifies the name of the OEM product to be converted:

`WEST`	specifies Westinghouse Disk Utility System
`DUSP`	specifies Blueline Disk Utility System
`MAXBACK`	specifies Blueline Maxback System
`FAVER`	specifies Faver System

`EOD=bbb` specifies the end of input delimiter statement.

The default delimiter is a /+ statement. You may specify any character string for the delimiter as long as:

- no imbedded blanks are included
- bbb doesn't start with '* $$'
- bbb doesn't exceed column 72

`EOJ=YES` specifies that a /& job control statement is to be added after the end of job (/*) is encountered for each job to be converted. 'NO' is the default.

`PWREOJ=YES` specifies that a '* $$ EOJ' Power JECL statement is to be added after the end of job (/*) is encountered for each job to be converted. 'NO' is the default.

`CMS=ccc` provides a conversion interface for VM/CMS users. CMS='FN FT' specifies that the conversion output on SYSPCH be preceded with a CMS :READ statement with a specific filename and filetype. The CMS filename and filetype specification must be enclosed within quotes.

`CMS=YES` will cause a CMS :READ statement to be generated for each job in the input stream. The filename generated will be extracted from each / / JOB statement encountered. The CMS filetype of 'JCL' will be used for each CMS file.

`CMS=POWER` performs the same function as CMS=YES except the filename will be extracted from each POWER * $$ JOB statement encountered.

`SAVE=LOGICAL` indicates that you wish to have SAVE=LOGICAL

	Dr.D statements generated for the backup / restore of VSAM files.
`TFNI=xxxxxxx`	specifies the TLBL filename for the input tape to be used.
`TFNO=xxxxxxx`	specifies the TLBL filename for the output tape to be used.

The following CMS commands will retrieve the SYSPCH output from the RDR and place it on CMS files:

`Order RDR nnn`	(place rdr file on tape of queue)
`Readcard *`	(build CMS files from rdr file)

Conversion Examples

The following examples illustrate various ways to convert existing jobs to Dr.D:

Example 1. Maxback conversion

This example illustrates the jobstream required to convert a Maxback jobstream to Dr.D:

```
// JOB CONVERT --- CONVERT MAXBACK TO DR.D ---
// ASSGN SYSPCH,PUNCH
// ASSGN SYS004,SYSIPT
// EXEC DRDCNVCU
CONVERT=MAXBACK,EOJ=YES,PWREOJ=YES
* ** JOB JNM=BUSYSRES,CLASS=A,DISP=L
* ** LST CLASS=A, DISP=L
// JOB MAXBAK --- BACKUP SYSRES TO TAPE ---
// ASSGN SYS000,TAPE
// ASSGN TLBL SYS000, 'SYSRES.BACKUP'
// EXEC MAXBACK
   SET LOG=ON,MTC=RUN,LIST=ON,CONFIRM=ON ;
   DUMP IJSYSRES(LIB) ;
/*
/+
/&
```

Example 2. Maxback conversion with unique EOD specification

This example illustrates the use of the EOD= specification

```
// JOB CONVERT --- CONVERT MAXBACK TO DR.D ---
// ASSGN SYSPCH,PUNCH
// ASSGN SYS004,SYSIPT
// EXEC DRDCNVCU
CONVERT=MAXBACK,EOJ=YES,PWREOJ=YES,EOD=%%
* ** JOB JNM=BUSYSRES,CLASS=A,DISP=L
* ** LST CLASS=A,DISP=L
// JOB MAXBAK --- BACKUP SYSRES TO TAPE ---
// ASSGN SYS000,TAPE
// TLBL SYS000,'SYSRES.BACKUP'
// EXEC MAXBACK
   SET LOG=ON,MTC=RUN,LIST=ON,CONFIRM=ON ;
   DUMP IJSYSRS(LIB) ;
/*
%%
/&
```

Example 3. Convert multiple Maxback jobs for a VM/CMS user

This example illustrates how to convert several jobs via a single execution of DRDCNVCU, returning the punched output to the CMS users reader queue. A CMS :READ statement will be generated for each job, using the * $$ JOB name as the filename:

```
* $$ JOB JNM=CONVERT,CLASS=A,DISP=D
* $$ PUN CLASS=V,DEST=(*,USER),JSEP=O
// JOB CONVERT --- CONVERT MULTIPLE MAXBACK JOBS ---
// ASSGN SYSPCH,PUNCH
// ASSGN SYS004,SYSIPT
// EXEC DRDCNVCU
CONVERT=MAXBACK,CMS=POWER
* ** JOB JNM=BUSYRES,CLASS=A,DISP=L
* ** LST CLASS=A,DISP=LUSYSRES,CLASS=A,DISP=L
* ** LST CLASS=A,DISP=L
// JOB MAXBAK --- BACKUP SYSRES TO TAPE ---
// ASSGN SYS001,TAPE
// TLBL SYS001,'SYSRES.BACKUP'
// EXEC MAXBACK
   SET LOG=ON,MTC=RUN,LIST=ON,CONFIRM=ON ;
   DUMP IJSYSRS(LIB) ;
/*
/&
* ** EOJ
* ** JOB JNM=RSSYSRES,CLASS=A,DISP=L
* ** LST CLASS=A,DISP=L
// JOB MAXBAK --- RESTORE SYSRES FROM TAPE ---
// ASSGN SYS000,TAPE
// TLBL SYS000,'SYSRES.BACKUP'
// EXEC MAXBACK
   SET LOG=ON,MTC=RUN,LIST=ON,CONFIRM=ON ;
   RESTORE IJSYSRS(LIB) ;
/*
/&
* ** EOJ
/+
/&
* $$ EOJ
```

Example 4.Convert Westinghouse Dump/Restore job.

This example illustrates the conversion of a Westinghouse job specifying the Convert control statement via the / / exec statement Parm :

```
// JOB CONVERT --- CONVERT WESTINGHOUSE TO DR.D ---
// ASSGN SYSPCH,PUNCH
// ASSGN SYS004,SYSIPT
// EXEC DRDCNVCU,PARM='CONVERT=WEST,EOJ=YES,PWREOJ=YES'
* ** JOB JNM=BUSYSRES,CLASS=A,DISP=L
* ** LST CLASS=A,DISP=L
// JOB WESTI --- BACKUP SYSRES TO TAPE ---
// ASSGN SYS000,TAPE
// ASSGN SYS003,DISK,VOL=DOSRES,SHR
// TLBL SYS000,'SYSRES.BACKUP'
// EXEC COPYDT
   IJSYSRES.SD
/*
/+
/&
```

Example 5. Convert Faver JCL

This example illustrates the conversion of a Faver Restore job specifying the Convert control statement via the / / EXEC statement Parm:

```
// JOB CONVERT --- CONVERT FAVER TO DR.D ---
// ASSGN SYSPCH,PUNCH
// ASSGN SYS004,SYSIPT
// EXEC DRDCNVCU,PARM='CONVERT=FAVER,EOJ=YES,PWREOJ=YES'
* ** JOB JNM=FAVER,CLASS=0,DISP=L
* ** LST CLASS=A,DISP=D
// JOB FAVER
// ASSGN SYS000,180
// DLBL TESTFLE,'VSAM.TEST.FILE',,VSAM
// TLBL FVRIN,'VSAM.BACKUP'
// EXEC GVRESTOR,SIZE=AUTO
RESTORE
CATDEF=USER.CATALOG.ONE
CLUSTER
CL=CICS.TEMP.STOR
CLDD=TESTFLE
/*
/+
/&
```

Appendix X - Dr.D Messages

The following error and write-to-operator messages are issued by Dr.D. Each message includes the probable reason for the message, any action recommended, and the disposition. The messages are listed by the two- or three-digit DOCTOR number. If return code processing is in effect, (see the RP= keyword), then any message that would cause cancellation will cause termination with return code 128. Any other message will cause termination with a return code of 1-7 depending on the severity of the condition causing the message. The return code is printed on each message number line where applicable.

DOCTOR 00 ID='XXX...XX NOT ON SYSXXX=CUU **RC128**

Where: XXX...XXX is the file identification

Cause: The stated file identification is not in the VTOC of the disk on the unit. This file-id is obtained either explicitly through the 'SIDENT=' or 'CIDENT' or 'RIDENT=' operands or from the DLBL card pointed to by a default or explicit file name ('SFILE=' or 'RFILE=').

Action: Correct the control statement or DLBL card and rerun the job.

Dispo: The job is canceled.

DOCTOR 01 INVALID KEYWORD ON CTLCRD XXXXX **RC128**

Where: XXXXX is an unidentifiable keyword 'XXXXX=' on the control statement.

Cause: XXXXX is an invalid keyword.

Action: Please refer to Appendix I for the appropriate keyword to use.

Dispo: The job is canceled.

DOCTOR 02 INVALID OPERAND SAVE/RESTORE=XXX **RC128**

Where: XXX was the subject of a 'SAVE=', 'COPY=', or 'RESTORE=' operand on the control statement.

Cause: SAVE, RESTORE, or COPY was not one of the following.

Action: Please refer to Appendix VII for summary of allowable SAVE, RESTORE, and COPY commands.

Dispo: The job is canceled.

DOCTOR 03 INVALID OPERAND SYSNO=XXXXXX **RC128**

Where: XXXXXX is the subject of a 'SSYSNO=', 'CSYSNO=', or 'RSYSNO=' on the control statement.

Cause: The specified SYS number was either non-numeric or not one of the following:

SYSRDR	SYSSLB
SYSIPT	SYSRLB
SYSPCH	SYSUSE
SYSLST	SYSREC
SYSLOG	SYSCLB
SYSLNK	SYSVIS
SYSRES	SYSCAT

Action: None.

Dispo: The job is canceled.

DOCTOR 04 SYSXXX IS ASSIGNED IGN, UA, OR NON-DASD **RC128**

Where: SYSXXX is either a default or explicit SYS number.

Cause: Dr.D has determined that the SYS number of the disk to be saved/copied/restored is either assigned UA, IGN, or a non-disk type device.

Action: Correct the assignment and rerun the job.

Dispo: The job is canceled.

DOCTOR 05 REST=PUTLABEL BUT FILXXXX NOT SD OR DA **RC128**

Where: FILXXXX was the filename specified in 'RF='.
Cause: RE=PU was specified, but the file's organization is not SD or DA.
Action: None.
Dispo: The job is canceled.

DOCTOR 06 NO DLBL/EXTENTS FOR FILXXXX **RC128**

Where: FILXXXX is the filename subject of either an 'SF=' or 'RF=' operand of the control statement.
Cause: There are no DLBL/EXTENT cards either in JCL or standard labels for the file.
Dispo: The job is canceled.

DOCTOR 07 FILE NAME UNSPECIFIED FOR SAVE/RESTORE=XXX **RC128**

Where: XXX is DATA, REORG, or PS, PR, PC, PP
Cause: For SAVE=DATA 'SFILE=' must be specified
RESTORE=DATA, 'RFILE=' must be specified
COPY=DATA, 'SF=' and 'RF=' must be specified
COPY/RESTORE=REORG 'RF=' must be specified
COPY/RESTORE=PS, PR, PC and LIB=NEW, 'RF=' must be specified.
Dispo: The job is canceled.

DOCTOR 08 RESTORING VOLUME FROM A XXXX TO A YYYY **RC128**

Where: XXXX is the save disk type, YYYY is the restore disk type
Cause: The saving and restoring disk types differ for volume SAVE/REST/COPY (i.e. A 3375 was saved, and is being restored to a 3380).
Dispo: The job is canceled.

DOCTOR 09 INVALID OPERAND SAVE=XXX,BUT RESTORE=YYY **RC128**

Where: XXX was the subject of 'SAVE=' YYY is the subject of 'RESTORE=' on the control statement.
Cause: An attempt is made to restore one type of save as another type of restore, (i.e. Save=PR but RESTORE=PS, or SAVE=APP but RESTORE=DATA).
Dispo: The job is canceled.

DOCTOR 10 RESTORING FILXXXX OVERFLOWS ALLOCATED DISK AREA **RC128**

Where: FILXXXX is a default or explicit file name
Cause: For COPY/RESTORE=PH Dr.D has exceeded the 'RH=' disk address. For COPY/RESTORE=DA, Dr.D has exceeded the allocated extent (SD or DA).
Dispo: The job is canceled.

DOCTOR 11 INVALID OPERAND RMODE=X RC128

Where: X is the operand of 'RMODE=', specified on the control statement.
Cause: RMODE may either be 'I' for image, 'S' for squeeze, 'U' for update, or 'A' for addon.
Dispo: The job is canceled.

DOCTOR 12 INVALID PHYSICAL ADDRESSES CCCHH,CCCHH FOR SYSXXX=CUU A YYYY **RC128**

Where: CCCHH are the extent start and stop addresses where CCC the cylinder and HH is the head number SYSXXX is the default or explicit SYS number CUU is the device address pointed to by the SYS number YYYY is the type of disk, either a:
2311 for a 2311 disk
2314 for a 2314 or 2319 disk
3330 for a 3330 disk
334a for a 3340 m35 disk
334b for a 3340 m70 disk
3350 for a 3350 disk
3375 for a 3375 disk

3380 for a 3380 disk
FBA for an FBA disk

Cause: Either you have exceeded the cylinder or head capacity of the particular type of disk drive, or the ending address physically comes before the starting address.

Dispo: The job is canceled.

DOCTOR 13 PUTLABEL SPECIFIED IN COPY/SAVE MODE **RC128**

Cause: 'COPY=PUTLABEL' or 'SAVE=PUTLABEL' was specified on the control statement.

Action: Correct control statement to read 'RESTORE=PU'.

Dispo: The job is canceled.

DOCTOR 15 ERASE DISK SPECIFIED IN COPY/SAVE MODE **RC128**

Cause: SAVE=ERASE or COPY=ERASE was specified on the control statement.

Action: Correct to RESTORE=ERASE.

Dispo: The job is canceled.

DOCTOR 16 SAVING FROM/RESTORING TO PACK VOL SER NO NNNNNN SYSXXX=CUU **RC128**

Where: NNNNNN is the volume serial number, SYSXXX is the default or explicit SYS number, CUU is the device address pointed to by the SYS number.

Cause: Dr.D will gain information about a particular SAVE/COPY/ RESTORE either by accessing the information itself or letting LIOCS do it by performing open/close routines. Examples of the first are volumes, and some library and miscellaneous functions. Examples of the latter are data files and some types of library save/copy/restores. Whenever the control information is garnered in the former fashion, this message is issued to confirm your intentions. For the latter types of access, the normal DOS/LIOCS procedures and messages insure against errors and this message is not issued. Action: If the correct pack is in the correct drive, hit the EOB key to continue or else type 'C' to cancel Dr.D.

DOCTOR 17 LINKED PHASE EXCEEDS LIMIT* TAPE NOT IPLABLE! **RC128**

Cause: During SAVE=IPL, the standalone supervisor and phases exceeds the size limit.

Action: Check that the tape blocksize is within limits. If it is not within limits, contact product support.

Dispo: The job is canceled.

DOCTOR 18 ERROR ON IPL CREATION * TAPE NOT IPLABLE! **RC128**

Cause: During a 'SAVE=IPL' a questionable condition has occurred during the tape I/O. Since it is imperative that the beginning portion of the tape (IPL records) be error-free (no ERP during IPL), the slightest aberration from the norm will prompt this message.

Dispo: The job is canceled.

DOCTOR 19 CORE DUMP DESIRED

Cause: A questionable condition has occurred while running in the diagnostic mode.

Action: Reply 'Y' to obtain a core dump of the Dr.D partition or else reply with an 'N'.

DOCTOR 20 EXECUTION TO RESUME **RC001**

Cause: A questionable condition has occurred during a DOCTOR execution requiring an operator decision.

Action: Reply 'Y' to ignore the error, 'I' to ignore the error and any other out of sequence errors that may occur for that file, or 'N' to cancel.

DOCTOR 21 RECORD DUMP DESIRED **RC001**

Cause: An ISAM sequence check or duplicate record condition has occurred.

Action: Reply 'Y' if current record is to be displayed, otherwise type an 'N'.

DOCTOR 22 PAGE FIX STAT IN BYTES XXXXX FIXED YYYYY NOT FIXED

Where: XXXXX represents the total number of fixed decimal bytes. YYYYY represents the total number of bytes not fixed in decimal. The sum of XXXXX and YYYYY is the total number of bytes Dr.D attempted to fix.

Cause: When executing in the virtual mode with PFIX support in the supervisor Dr.D will attempt to fix the above mentioned portions in real storage. When there is insufficient real core for this, the message is logged.

Dispo: As many as possible pages are fixed and the job continues.

DOCTOR 23 SYSXXX IS ASSIGNED IGN OR UA RC128

Where: SYSXXX is a default or explicit SYS number

Cause: While trying to set DASD file protect limits for a particular SYS number, Dr.D has determined that the SYS number is assigned UA or IGN.

Action: Make the correct assignment and rerun the job.

Dispo: The job is canceled.

DOCTOR 24 SYSXXX IS ASSIGNED TO NON-DASD RC128

Where: SYSXXX is a default or explicit SYS number

Cause: While trying to set DASD file protect limits for a particular SYS number, Dr.D has determined that the SYS number is assigned to a non-disk type device.

Action: Make the correct assignment and rerun the job.

Dispo: The job is canceled.

DOCTOR 26 NO MATCH ON SYSRES END CYL, ALLOC=XXX BUT LABEL=YYY RC128

Where: XXX is the sum of the specified or default library allocations plus begin cylinder (sub-directories, etc.) plus end cylinder (label area). YYY is the number of cylinders as computed from the extent card for the filename supplied in 'RF='.

Cause: The allocated number of cylinders exceed the number of cylinders from label information.

Dispo: The job is canceled.

DOCTOR 27 INVALID ALLOCATIONS CL=CCCTT RL=CCCTT SL=CCCTT PL=CCC RC128
FBA type disk use code-> BBBBBBDDD in place of CCCTT

Where: CCCTT are the allocations for CORE IMAGE, RELO, SOURCE, and PROCEDURE libraries respectively. CCC is the number of cylinders in the library. TT is the number of directory tracks. BBBBBBDDD is FBA library & directory blocks.

Cause: CCCTT/BBBBBBDDD was neither 'S', or numeric, or the core image allocation was zero.

Dispo: The job is canceled.

DOCTOR 28 PHYSICAL DISK ADDRESS MISSING/INVALID RC128

Cause: A function requiring the entry of disk start and end address is being performed and a required disk address is invalid or not entered. A disk address may be invalid for three reasons: 1) the start address exceeds the end address, 2) the end address exceeds the device limit, 3) for CKD disks the head number is not within assumed/specified limits.

Dispo: The job is canceled.

DOCTOR 30 REST=XXX BUT FILXXXX NOT ON CYL BOUNDARY FILE START=CCCHH,STOP=CCHH RC128

Where: XXX is the subject of the 'COPY/RESTORE=' operand FILXXXX is the default or explicit filename of a private library or SYSTEM residence file. CCCHH represents the file start and stop extents. CCC is the cylinder. HH is the head.

Cause: When copying/restoring either a private library (PS, PR, PC, PP) or a SYSTEM residence file (reorg), the files must be on a cylinder boundary.

Dispo: The job is canceled.

DOCTOR 31 DIRTRKS=XX NOT NUMERIC RC128

Where: XX is the subject of the 'DIRTRKS=' operand

Cause: The specified 2-digit directory tracks is non-numeric or zero. Or the specified 3 digit directory blocks is non-numeric or zero.

Dispo: The job is canceled.

DOCTOR 32 SAVE/REST=XXX ON SYSXXX=CUU BUT LIBRARY NON-EXISTENT RC128

Where: XXX is PS, PR, PC, PP, SS, SR, SC, SP SYSXXX is the default or specified SYS number. CUU is the device address pointed to by SYSXXX.

Cause: 'REST/COPY=' with 'LIB=O' or 'LIB=A' or 'SAVE=' was specified for a library, but the target library is either non-existent or not of the type specified.

Dispo: The job is canceled.

DOCTOR 33 LIBRARY=X INVALID OPERAND RC128

Where: X is the subject of 'LIB=' on the control statement.

Cause: 'LIB=' was neither 'N' (new), 'O' (old), 'A' (add), nor 'C' (create SP-n) nor 'E' (extend SP-n).

Dispo: The job is canceled.

DOCTOR 34 RESTORE=XXX BUT LIBRARY=NEW INVALID FOR A SYSTEM LIBRARY

Where: XXX is either SC, SP, SR, SS.

Cause: 'LIB=N' is invalid when restoring a SYSTEM library.

Action: To create a new SYSTEM library, use the 'REORG' function.

Dispo: The job is canceled.

DOCTOR 35 ENTER RESTORE DISK TYPE

Cause: During a standalone restore Dr.D needs to know the type of restore disk.

Action: Respond with any of the following:
2311 for 2311
2314 for 2314, 2319
3330 for 3330
334A for 3340 M35
334B for 3340 M70
3350 for 3350
3375 for 3375
3380 for 3380
3390 for 3390
FBA for FBA

DOCTOR 36 ENTER RESTORE DISK CUU

Cause: During a standalone restore, Dr.D needs to know the device address of the restore disk.

Action: Respond with the 3-digit channel and unit number of the disk drive. If CUU,VOLSER is entered, the restored disk will have its volser changed to the 'VOLSER' value.

DOCTOR 37 PROGRAM CHECK

Cause: A software error has occurred during a standalone restore.

Action: Rerun. If the problem recurs take a standalone core dump and contact product support.

Dispo: The job enters permanent wait state.

DOCTOR 38 MACHINE CHECK

Cause: An unexpected machine check has occurred during a standalone restore.

Action: Rerun the job.

Dispo: The job enters permanent wait state.

DOCTOR 39 UNRECOVERABLE ERROR ON TAPE

Cause: A tape read error occurred during a standalone restore.

Action: Clean the tape drive or use another drive and rerun.

Dispo: The job enters permanent wait state.

DOCTOR 40 VOLUME RESTORE COMPLETED-TYPE END OR HIT ENTER TO CONTINUE

Cause: The standalone restore of a disk is completed.

Dispo: If more disks are to be restored, hit enter. If finished, type 'END'.

DOCTOR 41 UNRECOVERABLE ERROR ON DISK

Cause: A non-correctable disk I/O error has occurred during a restore.

Action: None.

Dispo: The job is canceled.

DOCTOR 42 IPL SPECIFICATION IN SAVE/COPY MODE

Cause: 'RESTORE=IPL' or 'COPY=IPL' was specified on the control statement.

Action: Correct control statement to 'SAVE=IPL'.

Dispo: The job is canceled.

DOCTOR 43 UNABLE TO FIND SPECIFIED FILE **RC128**

Cause: During restore operation, the specified file name could not be found on the save tape. Or the name is incorrect, precedes the point at which searching began, or the wrong tape is mounted. If doing a 'RESTORE=LOGICAL' operation check to see if 'SCAT=' needs to be added when restoring to a catalog other than that saved, or that the 'SVOLSER=' parameter is correct for non-VSAM.

Action: None.

Dispo: The job is canceled.

DOCTOR 44 CYLOFLOW=XX INVALID OPERAND **RC128**

Where: XX is the subject of the 'CYL=' operand.

Cause: The specified 2 digit cylinder overflow is either non-numeric or exceeds the capacity for the particular restoring disk:

DISK	MAXIMUM CYLOFLO
2311	08
2314	18
3330	17
3340	10

Action: None.

Dispo: The job is canceled.

DOCTOR 45 NRECDS=XX INVALID OPERAND **RC128**

Where: XX is the subject of the 'NRECDS=' operand.

Cause: The specified 'NRECDS=' blocking factor is either zero or non-numeric.

Action: None.

Dispo: The job is canceled.

DOCTOR 46 DELETE CODE=PPPPHH INVALID OPERAND **RC128**

Where: PPPPHH is the subject of the 'DELETE=' operand. PPPP is a 4-digit position number (relative to one). HH is a 2-digit position field representing the hexadecimal form of the delete code.

Cause: The specified position PPPP was non-numeric, 0, or greater than 4096 or HH wasn't one of the following: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

Action: None.

Dispo: The job is canceled.

DOCTOR 47 CARDCOUNT=XX INVALID OPERAND **RC128**

Where: XX is the subject of the 'CARDCOUNT=' operand on the control statement.

Cause: Either the 2-digit 'CARDCOUNT=' operand was non-numeric or zero, or the operand appeared twice during the execution of Dr.D.

Action: None.
Dispo: The job is canceled.

DOCTOR 48 CARDCOUNT=XX EXCEEDED **RC128**

Where: XX is the subject of the 'CARDCOUNT=' operand on a submitted control statement.
Cause: XX number of cards was specified to be in the JCL, but Dr.D read the XX+1 card.
Dispo: The job is canceled.

DOCTOR 49 WARNING! CARDS MISSING FROM JCL CARDCOUNT SHOULD BE XX BUT WAS YY **RC128**

Where: XX is the subject of the 'CARDCOUNT=' operand on a submitted control statement. YY is the actual number of cards read since and including the card with the 'CARDCOUNT=' operand on it until, but not including the /*.
Cause: The actual count YY is smaller than it should be.
Action: Replace YY with XX.
Dispo: The job is canceled.

DOCTOR 50 SEQNO=XX INVALID OPERAND **RC128**

Where: XX is the subject of the 'SEQNO=' operand on the control statement.
Cause: The 2-digit sequence number is either zero or non-numeric.
Action: None.
Dispo: The job is canceled.

DOCTOR 51 SEQNO=XX BUT FILE IS ACTUALLY 'YY' ON INPUT/OUTPUT **RC128**

WHERE: XX is the subject of the 'SEQNO=' and operand on the control statement. YY is the sequence number internally generated by Dr.D.
Cause: The specified and actual sequence numbers do not match. For restores this indicates a tape positioning problem or for saves/copies this indicates one or more control cards out of sequence.
Action: None.
Dispo: The job is canceled.

DOCTOR 52 WRONG LENGTH RECORD ON TRACK CYL=CCC HEAD=HH ON CUU **RC002**

Where: CCC is the cylinder number. HH is the head number. CUU is the disk device address.
Cause: When saving/copying a library, Dr.D has encountered a track with bad records on it. It is imperative that the track at least be properly formatted immediately, in case DOS accesses this area. (See miscellaneous functions - physical track copy).
Action: None.
Dispo: The member(s) of the library that resides on this track is no good. However, the save tape has been formatted with dummy data so that this track may be restored okay. The library SAVE then continues.

DOCTOR 53 TAPE=XXX INVALID OPERAND **RC128**

Where: XXX is the subject of a 'TAPE=' operand on a control statement.
Cause: The specified tape parameter was not acceptable. The allowable parameters are REW, RUN, NOR, for tape control; or F1, F2, F3, F4, and F5 to specify ALIEN tape format; or a numeric value to specify the number of tapes to be processed. Refer to the keyword section for more information.
Action: Correct the parameter and resubmit the job.
Dispo: The job is canceled.

DOCTOR 54 POSITION=XXX INVALID OPERAND **RC128**

Where: XXX is the subject of the 'POSIT=' operand on a control statement.
Cause: The specified 3-digit position operand was either zero, non-numeric, or less than the current file number on the restore tape (backward positioning is not allowed).
Action: None.

Dispo: The job is canceled.

DOCTOR 55 SKIP=XXX INVALID OPERAND **RC128**

Where: XXX is the subject of the 'SKIP=' operand on the control statement.
Cause: The 3-digit specified skip operand was non-numeric.
Action: None.
Dispo: The job is canceled.

DOCTOR 56 ISAM RESTORE FILXXXX DASD ERROR at MBBCCHHR=DDDDDDDDDDDDDDDD **RC128**

Where: FILXXXX is the filename of the restored/copied ISAM file DDD...DD is the 8-byte hexadecimal record ID of the form

```
 M  =  EXTENT SEQUENCE NUMBER
BB  =  BIN NUMBER
CC  =  CYLINDER NUMBER
HH  =  HEAD NUMBER
 R  =  RECORD NUMBER
```

Cause: ISAM LIOCS detected a disk error at the specified location.
Action: Rerun the job.
Dispo: The job is canceled.

DOCTOR 57 ISAM RESTORE FILXXXX WRONG LENGTH RECORD AT MBBCCHHR=DDDDDDDDDDDDDDDD **RC128**

Where: FILXXXX is the filename of the restored/copied ISAM file DDD...DDD is the 8-byte hexadecimal record ID of the form

```
 M  =  EXTENT SEQUENCE NUMBER
BB  =  BIN NUMBER
CC  =  CYLINDER NUMBER
HH  =  HEAD NUMBER
 R  =  RECORD NUMBER
```

Cause: ISAM LIOCS detected a wrong length record at the specified location.
Action: Rerun the job.
Dispo: The job is canceled.

DOCTOR 58 ISAM RESTORE FILXXXX PRIME DATA AREA FULL **RC128**

Where: FILXXXX is the filename of the restored/copied ISAM file.
Cause: The prime data area is too small to contain all of the file.
Action: Enlarge prime data area and rerun the job.
Dispo: The job is canceled.

DOCTOR 59 ISAM RESTORE FILXXXX CYLINDER INDEX AREA FULL **RC128**

Where: FILXXXX is the filename of the restored/copied ISAM file.
Cause: The cylinder index area is too small to contain all of cylinder indexes.
Action: Enlarge the cylinder index area extent and rerun the job.
Dispo: The job is canceled.

DOCTOR 60 ISAM RESTORE FILXXXX MASTER INDEX AREA FULL **RC128**

Where: FILXXXX is the filename of the restored/copied ISAM file.
Cause: The master index is too small to hold all the master index records.
Action: Enlarge master index extent and rerun the job.
Dispo: The job is canceled.

DOCTOR 61 ISAM RESTORE FILXXXX PRIME DATA AREA OVERFLOW **RC128**

Where: FILXXXX is the filename of the restored/copied ISAM file.
Cause: The prime data area is just one track too small. (It can't contain the EOF record).
Action: Enlarge the prime data area extent and rerun the job.

Dispo: The job is canceled.

DOCTOR 62 ISAM RESTORE FILXXXX DUPLICATE RECORD

Where: FILXXXX is the filename of the restored/copied ISAM file.
Cause: A duplicate key was encountered when restoring or copying an ISAM file.
Action: None.
Dispo: Messages 21 and 20 follow.

DOCTOR 63 ISAM RESTORE FILXXXX SEQUENCE CHECK

Where: FILXXXX is the filename of the restored/copied ISAM file.
Cause: An out-of-sequence key was encountered while restoring or copying an ISAM file.
Action: None.
Dispo: Messages 21 and 20 follow.

DOCTOR 64 BLOCK SIZE=XXXXX INVALID OPERAND **RC128**

Where: XXXXX is the blocksize specified as the subject of the 'BLOCKSIZE=' operand.
Cause: The 5-digit blocksize was non-numeric.
Action: None.
Dispo: The job is canceled.

DOCTOR 65 UNPAIRED QUOTE ON CONTROL CARD **RC128**

Cause: A '/*' was encountered while scanning for the ending quote mark on the control statements.
Action: Eliminate extraneous '/*'.
Dispo: The job is canceled.

DOCTOR 66 SAVE TAPE CREATED ON DD/DD/DD AT TT-TT-TT **RC128**

Where: DD/DD/DD is the date. TT-TT-TT is the time.
Cause: When restoring, the date and time that the save tape was created is logged out.
Action: Hit ENTER/EOB to continue, type 'C' to cancel Dr.D.

DOCTOR 67 MEMBERLIST=X INVALID OPERAND **RC128**

Where: X is either 'S' for select or 'R' for reject.
Cause: The specified operand is neither 'S' nor 'R'.
Action: None.
Dispo: The job is canceled.

DOCTOR 68 PARTITION TOO SMALL FOR MEMBERLIST **RC128**

Cause: There is insufficient core for the member list.
Action: 'ALLOC' the partition with the next 2K increment in size and rerun job.
Dispo: The job is canceled.

DOCTOR 69 SAVING FROM/ PACK VOL SER NO NNNNNN SYSXXX=CUU MOUNT MMMMMM restoring to. **RC128**

Where: NNNNNN is the volume serial number of the mounted disk. SYSXXX is the default or explicit SYS number. CUU is the device address pointed to by the SYS number. MMMMMM is the volume serial number of the requested disk.
Cause: SVOLID, RVOLID, or CVOLID=MMMMMM was specified on a control card but the pack is not mounted.
Action: Mount the right pack on the right drive, hit EOB to continue, or else type 'C' to cancel Dr.D.

DOCTOR 70 ISAM DLBL FOR XXXXXXX NOT ISC **RC128**

Where: XXXXXXX is the filename specified in 'RF='
Cause: 'RESTORE=DATA' is specified but the DLBL type for the file is ISE.

Action: Correct the DLBL card to read ISC and rerun the job.
Dispo: The job is canceled.

DOCTOR 71 RTRACK/STRACK=NN NON-NUMERIC **RC128**

Where: NN is the object of a 'RT/ST/CT=OPERAND'
Cause: The specified two decimal digit lower head limit is non-numeric.
Action: None.
Dispo: The job is canceled.

DOCTOR 72 RUPPER/SUPPER=NN NON-NUMERIC **RC128**

Where: NN is the object of a 'RU/SU/CU=OPERAND'
Cause: The specified two decimal digit upper head limit is non-numeric.
Action: None.
Dispo: The job is canceled.

DOCTOR 73 INVALID OPERAND R/STRACK=NN R/SUPPER=MM FOR SYSXXX=CUU A DDDD **RC128**

Where: NN is the subject of a 'RT/ST/CT=OPERAND' MM is the subject of a 'RU/SU/CU=OPERAND' SYSXXX is a default or explicit SYS number. CUU is the device address pointed to by the SYS number. DDDD is the type of disk:

either a 2311 for 2311
2314 for 2314, 2319
3330 for 3330
334A for 3340 M35
or 334B for 3340 M70
3350 for 3350
3375 for 3375
3380 for 3380
3390 for 3390

Cause: During a 'SAVE/RESTORE/COPY=PHY', you have either 1) specified an upper or lower head limit that exceeds the capacity of the disk, or 2) your specified lower head limit exceeds the upper head limit.
Action: None.
Dispo: The job is canceled.

DOCTOR 74 LIBRARY FILLED UP WITH MEMBER=XXXXXXXX **RC002**

Where: XXXXXXXX is the name of a member of the restore library. YYY is the subject of a 'RESTORE=' or 'COPY=' operand on the control statement.
Cause: While restoring/copying the indicated library, Dr.D ran out of room in the library area. The indicated member was the first one that could not fit.
Action: Enlarge the library and rerun the restore/copy process.
Dispo: The restore/copy continues. However, the indicated member and all following members are not restored or copied.

DOCTOR 75 DIRECTORY FILLED UP WITH MEMBER=XXXXXXXX **RC002**

Where: XXXXXXXX is the name of a member of the restore library. YYY is the subject of a 'RESTORE=' or 'COPY=' operand on the control statement.
Cause: While restoring/copying the indicated library, Dr.D ran out of directory tracks. The indicated member was the first one that couldn't fit.
Action: Enlarge the directory and rerun.
Dispo: The restore/copy continues. However, the indicated member and all following members are not restored or copied.

DOCTOR 76 RECORD RRR ON TRACK CYL-CCC HEAD-HH THINKS IT IS ON CYL-XXX HEAD YY

Where: RRR is the record number. CCCHH is the cylinder and head number of the track the records reside on. XXX/YY is the cylinder and head number in the count field of the record. All fields are decimal.

Cause: While saving/copying a disk, Dr.D has encountered a record whose count area does not reflect the actual track it resides on.

Action: None.

Dispo: While the incorrect count area remains on the save disk, the correct count area is reflected on the save tape or copied disk. The SAVE/COPY continues. NOTE: If you ditto-dump the bad track, you may or may not see the bad record(s).

DOCTOR 77 RESTORE TAPE NOT A Dr.D TAPE **RC128**

Cause: The tape input to Dr.D is not recognizable as a Dr.D tape. Either the wrong tape is mounted, or the right tape is positioned wrong.

Action: Mount the correct tape and hit enter or type cancel.

DOCTOR 78 INPUT BLOCKSIZE=XXXXX EXCEEDS Dr.D BLOCKSIZE=YYYYY **RC128**

Where: XXXXX is the blocksize of the input tape. YYYYY is the current blocksize of the Dr.D program.

Cause: An input file's blocksize is larger than Dr.D's buffer.

Action: Rerun with a 'BLOCKSIZE=' parameter on the control card to enlarge the buffer.

Dispo: The job is canceled.

DOCTOR 79 CTL FILE NOT PRESENT **RC128**

Cause: During either a save or restore, a 'DSN=' parameter was specified. However, Dr.D cannot locate the control file due to three reasons: 1) No DLBL/extents for file named 'Dr.D', 2) SYSNO on extent card not pointing to DASD, or 3) VTOC of disk does not contain file information.

Action: Reply 'I' to ignore and continue, or else reply 'C' to cancel.

DOCTOR 80 CTL ENTRY NOT PRESENT FOR DSN=XXXXXXXX **RC128**

Where: XXXXXXXX is the subject of a 'DSN=' operand

Cause: During a restore, a 'DSN=' was specified, but it has no corresponding entry in the control file.

Action: Reply 'I' to ignore and continue, or else reply 'C' to cancel.

DOCTOR 81 LAST SAVE FOR DSN=XXXXXXXX WAS ON DD/DD/DD AT TT-TT-TT ON TAPE NNNNN **RC128**

Where: XXXXXXXX is the subject of a 'DSN=' operand. DD/DD/DD is the date of the last save according to the control file. TT-TT-TT is the time of the last save according to the control file. NNNNNN is the volume serial number of the last save tape.

Cause: During a restore with 'DSN=' specified, the date and time of the last save disagrees with the date and time on the input tape.

Dispo: Reply 'I' to ignore and continue, or else reply 'C' to cancel.

DOCTOR 82 NO ROOM LEFT IN CTL FILE TO ENTER DSN=XXXXXXXX OF DISK **RC128**

Where: XXXXXXXX is the subject of a 'DSN=' operand.

Cause: During a save, a 'DSN=' was specified, but there is no room left in the control file.

Action: Enlarge the control file. Reply 'I' to ignore and continue, or else reply 'C' to cancel.

DOCTOR 83 RESTORE=REORG FILXXXX DOES NOT BEGIN ON FIRST TRACK **RC128**

Where: FILXXXX is the filename of the SYSRES file.

Cause: The file specified in the 'RF=' operand does not begin on relative track one of the CKD disk or block 2 of the FBA disk.

Action: Correct the extent card and rerun.

Dispo: The job is canceled.

DOCTOR 84 RESTORING ISAM/VSAM DATA TAPE PARAMETERS NON-NUMERIC DT=XXX.XXX **RC128**

Where: XXX...XXX is the 18-digit subject of a 'DT=' operand.

Cause: The specified data tape parameter is non-numeric.

Action: None.
Dispo: The job is canceled.

DOCTOR 85 DATE=XXXXX INVALID, NON-NUMERIC OR OMITTED **RC128**

Where: XXXXX is the subject of the 'DATE=' operand
Cause: During an 'RE=EXPIRDATE' function, the required date parameter was either omitted or invalid.
Action: Enter a 5-digit number of the Julian form YYDDD or a 'D' (for delete) or 'R' (for retain).
Dispo: The job is canceled.

DOCTOR 86 DISK READ ERROR **RC002**
DOCTOR 86 DISK READ ERROR IGNORED

Cause: A disk read error has occurred, or has occurred and has been ignored.
Action: None.

DOCTOR 87 CONTROL CARD TABLE OVERFLOW **RC128**

Cause: UPSI bit 7 (/ / UPSI XXXXXX1) is on and more control cards have been entered than will fit in memory.
Action: None.
Dispo: The job is canceled.

DOCTOR 88 CONTROL CARD HAS NEITHER FIND= OR RF= **RC128**

Cause: UPSI bit 7 (/ / UPSI XXXXXX1) is on and a control card was entered that had neither the FIND= or RF= PARAMETER
Action: None.
Dispo: The job is canceled.

DOCTOR 89 Dr.D SOFTWARE ERROR **RC128**

Cause: Dr.D has detected an illogical condition during processing which it assumes to be an error it has created.
Action: If option dump was not in effect, re-submit the job with the dump option. A memory dump will be produced (SVC 255). Please contact your Dr.D representative.

DOCTOR 90 WRONG LENGTH RECORD AT CCCHHR (HEX)=CCCHHR **RC002**

Cause: A wrong length record has been detected at the listed CCCHHR.
Action: None.

DOCTOR 91 ASSUMED/SPECIFIED BLOCK SIZE (AAAAA) LESS THAN BLOCK SIZE(BBBBB) found at cylinder CCCC, head HH, record RRR. **RC128**

Cause: Dr.D has encountered a physical block greater than the assumed or specified blocksize.
Action: Increase the blocksize to at least track capacity and rerun (use BL=). In case of some disk errors, a count area may be read that indicates a physical block exceeding track capacity, usually in an unused area of the disk. The erase function can in some cases be used to clear the bad count area.
Dispo: The job is canceled.

DOCTOR 92 LOGICAL RECORD TOO LARGE FOR SPECIFIED BLOCK SIZE RECORD LENGTH **RC128**

Cause: A logical record has been read that is larger than the assumed or specified blocksize.
Action: Increase the blocksize (see BL parameter) and rerun the job.
Dispo: The job is canceled.

DOCTOR 93 BLOCK SIZE NOT EVENLY DIVISIBLE BY SPECIFIED LOGICAL **RC128**

Cause: The logical record length was specified and the file block size is not evenly divisible by it.
Action: None.

Dispo: The job is canceled.

DOCTOR 94 UPSI XXXXXX1X IS SPECIFIED, BUT SAVE AND RESTORE BOTH SPECIFIED RC128

Cause: The UPSI setting indicates that a 'MATCHING RESTORE' is to be performed, but a mixture of restore/save/ copy control cards have been entered.

Action: None.

Dispo: The job is canceled.

DOCTOR 95 LIBR UNABLE TO DEFINE SPECIFIED LIBRARY RC128

Cause: The IBM LIBR program has been called to define an SP2 library when LIB=NEW has been specified, and the library cannot be defined.

Action: Review LIBR output on SYSLST.

Dispo: The job is canceled.

DOCTOR 96 SP2 LIBRARY DISK ERROR/INVALID LIBRARY FORMAT RC128

Cause: An unformatted track has been detected or an unrecoverable disk error has occurred within an area of the library being read or written.

Action: Run the IBM LIBR test function to test the library for errors

Dispo: The job is canceled.

DOCTOR 97 DIRECTORY ERROR PROCESSING SUB-LIBRARY SSSSSSSS RC002

Cause: During VSE-SP library backup, an error in the directory has caused processing of the sub-library listed to be aborted.

Action: Run the LIBR test function to determine library condition.

Dispo: Backup continues with the next sub-library, if possible.

DOCTOR 98 BACKUP NOT SAVE ALL/CUR OR WRONG DEVICE TYPE

Cause: During standalone restore, DR.D has found tape data not consistent with the required backup type or device type. This could be caused by tape errors, wrong tape mounted, etc.

Action: Make sure that the correct tapes are mounted in the right order. If the problem persists, try using different tape drives. If the VSE system is operational, try the restore under VSE. If a VM operating system is available, try the standalone under VM.

DOCTOR 99 INADEQUATE MEMORY ALLOCATED TO PARTITION RC128

Cause: Not enough partition memory is available to allow allocation of buffers & tables for specified function.

Action: Allocate more partition memory size either by use of the appropriate allocate command or the SIZE= PARAMETER on the EXEC control card.

DOCTOR 100 SAVE/RESTORE TYP=XXX PARM1=AAA PARM2=BBB PARM3=CCC..etc.

Where: TYP=XXX is one of the following:

VOL=ALL	LIB=SC
VOL=CUR	LIB=SR
DATA=DA	LIB=SS
DATA=SD	LIB=SP
DATA=ISAM	LIB=PC
DATA=VSAM	LIB=PR
DATA=LOGICAL	LIB=PS
PHYSICAL	LIB=PP
REORG	LIB=LI

PARMX=YYY is one of the following:

(CYLOFLOW=)	(CYLINDER OVERFLOW)	
NRECDS=	BLOCKING FACTOR	*V
KEYLEN=	KEY LENGTH	*S
DATALEN=	DATA LENGTH	*A

LOG RECDS RD/WRTTN=	NUMBER LOGICAL RECORDSREAD/WRITTEN	*M
LOG RECDS DELETED=	NUMBER LOGICAL RECORDS DELETED	**
BLKS RD/WRTTN=	NUMBER OF PHYSICAL BLOCKS READ	*PAC
	OR WRITTEN (INCLUDING EOF RECDS)	*HLU
EOF=CCC/HH/RRR ON CUU	THE DISK AND DEVICE ADDRESSES OF	*YLR
	THE FIRST EOF RECD READ/WRITTEN	*S
DIRTRKS=	NUMBER TRACKS IN DIRECTORY	
LIBCYL=	NUMBER CYLINDERS IN LIBRARY MEMBERS	
RD/WRTTN=	NUMBER OF MEMBERS READ OR WRITTEN	
ACTIVE BLKS RD/WRTTN=	NUMBER OF BLKS OCCUPIED BY MEMBERS	
DELETED BLKS READ=	NUMBER OF BLKS THAT CAN BE RECOVERED BY CONDENSING LIBRARY	
NUMBER SYNONYMS=	NUMBER OF MEMBERS REPLACED BY A LIB=A	
NUMBER SYNONYM BLKS=	NUMBER OF BLKS THEY OCCUPIED	
CL=LLLDD	CORE ALLOCATION LLL-LIBCYL	
RL=	RELO ALLOCATION DD -DIR TRKS	
SL=	SOURCE ALLOCATION	
PL=	PROC ALLOCATION	
LABCYL=	LABEL CYLINDER ADDRESS	

Cause: By specifying a '/ / UPSI XX1' in the job stream, various facts regarding the Dr.D execution will be written to SYSLOG. (If applicable MSG 22 regarding page fixing will also be logged). At the conclusion of a save, restore, or copy, various statistics pertinent to the operation just performed will be logged, starting with a message number of 100 and incrementing by one until complete.

Action: None.

Dispo: The job continues normally.

DOCTOR 106 UNABLE TO ACCESS LIBRARY/SUBLIBRARY **RC128**

Cause: Library or Sub-Library not accessible for restore by invoking LIBR functions. Possible JCL error or library locked by some other task.

Dispo: Check the JCL for the library and that no other task has exclusive control of the target library.

DOCTOR 107 UNABLE TO CREATE SUB-LIB **RC128**

Cause: Dr.D has invoked LIBR to create a new sub-library, but LIBR is unable to do so.

Dispo: Check JCL for library and that adequate space exists in the target library. See that no other task has exclusive control of the library.

DOCTOR 109 SPECIFIED BLOCK SIZE EXCEEDS TRACK CAPACITY **RC128**

Cause: While restoring a 'Data Tape' the specified Physical Block size exceeds the track capacity of the target disk. This blocksize is either the result of multiplying the logical record size by the number of records per block ('LR' times 'NR') or the maximum blocksize specified for variable blocked format ('LB').

Action: None.

Dispo: The job is canceled.

DOCTOR 110 BLOCK COUNT ERROR-BLOCKS READ=XXXXXXX, **RC002**
BLOCK CONTROL=XXXXXXX,FILE=FFFFFFF,'FILE ID'

Cause: While reading a backup tape, a block count error was detected in the file and file ID indicated.

Action: None.

Dispo: The operator has the option to cancel or continue. This is usually the result of a hardware failure.

DOCTOR 111 VSAM RECORD COUNT DISCREPANCY DATASET='DDDDDDD' SAVED=NNNNNN CATALOG=NNNNNN **RC001**

Cause: Fast VSAM backup saved record count disagrees with the catalog.

Dispo: Backup continues but user should investigate for cause.

DOCTOR 112 VSAM SEQUENCE ERROR **RC002**

Cause: Fast VSAM backup has detected a key out of sequence in a KSDS file as the result of file corruption or a possible Dr.D logic error.

Action: Retry the backup changing the 'FAST=YES' parameter to 'FAST=NO'. If the file saves successfully with 'FAST=NO', call product support.

DOCTOR 113 INCONSISTENT PARAMETERS SPECIFIED **RC128**

Cause: Parameters have been specified that do not logically go with the function being performed. An example would be specifying file type parameters with non-file type operations.

Action: None.

Dispo: The job is canceled.

DOCTOR 114 LIBRARY HAS OVER 16 EXTENTS **RC128**

Cause: The VSE-SP library create/extend feature has detected that the target library in VSAM space already has 16 or more extents, or that a non-VSAM library will have more than 16 extents after creation/extension.

Action: If the library is in VSAM space, a listcat will show how many extents exist. If the Library is in non-VSAM space, list VTOC and examine JCL EXTENT cards.

Dispo: The job is canceled.

DOCTOR 115 LIBRARY HAS NO ADDITIONAL EXTENTS **RC128**

Cause: The VSE-SP library extend feature has determined that target library cannot be extended.

Action: If the library is in VSAM space, a listcat will show how many extents exist. If the Library is in non-VSAM space, list VTOC and examine JCL EXTENT cards.

Dispo: The job is canceled.

DOCTOR 116 CONTROL INTERVAL FORMAT ERROR IN FILE (file id) **RC002**

Cause: During the backup of a VSAM file, one or more control intervals have been found to be in incorrect format possibly due to VSAM CI split processing.

Action: Check to see that the file is valid. If the file is usable, the backup can be used to restore the file, reorganizing and correcting any format errors.

Dispo: The backup continues.

DOCTOR 119 RMODE=UPDATE SPECIFIED BUT NO RECORD FOUND **RC255**

Cause: COPY/RESTORE=PHYSICAL WITH RM=U is specified, but no disk block has been found on the CKD disk at the specified cylinder-head-record.

Action: Check the disk address to see that it is correct. If it is, then print the disk track to insure that a block exists at that location.

Dispo: The job is canceled.

DOCTOR 120 NO AIX DD-NAME/NUMBER ENTERED FOR AIX BUILD FUNCTION (AF=)

Cause: FUNCTION=BUILD has been invoked, but no AF= parameter has been entered.

Action: Add the AF= parameter and rerun the job.

Dispo: The job is canceled.

DOCTOR 121 NEITHER SFNAME NOR RFNAME ENTERED FOR AIX BUILD FUNCTION

Cause: FUNCTION=BUILD has been invoked, but the base cluster has not been specified by either 'SF=' or 'RF='.

Action: Add the SF/RF= parameter(s) and rerun the job.

Dispo: The job is canceled.

DOCTOR 122 BUILD INDEX FUNCTION ABNORMAL TERMINATION ERROR CODE-XXX.

Cause: Abnormal termination of build aix function. The error code specifies the cause as follows:

002 Invalid cluster type.
003 Aix is not defined in VSAM catalog.
004 JCL or CATALOG error Check to see that 'AF=' parameter matches the DLBL for the AIX.
005 Build file specified not defined as AIX.
006 AIX open error.
007 The AF= parameter specifies more than 99 files.
008 Sort initialization error.
009 Invalid AIX key length.
010 GETVIS failed.
011 VSAM AIX write (put) error.
012 AIX maximum record length exceeded.
013 VSAM AIX MODCB failure.
014 VSAM AIX OPEN / CLOSE failure.
015 Product Expired
016 VSAM Catalog read error
017 VSAM Catalog write error
018 Aix key defined to small
019 Invalid KEY= control statement
020 Invalid UNIQUE= control statement
021 Invalid SELECT statement record field
022 Invalid SELECT statement expression
023 Invalid SELECT statement compare field
024 Invalid SELECT statement equation symbol
025 Select statement group compare not ended

Action: Fix indicated problem and rerun the job.

Dispo: The job is canceled.

DOCTOR 123 NONUNIQUE KEY FOUND FOR AIX=AAAAAAA

Cause: The VSAM AIX build function has found non-unique keys while building an AIX defined for unique keys.

Action: Check to see that the AIX is defined correctly.

Dispo: Processing continues, but the AIX is not created. If other AIX(s) are to be built, building continues with the next.

DOCTOR 124 MISSING AIX KEY FIELD, AIX=XXXXXXX

Cause: The VSAM AIX build function has read a logical record that is too short to contain the specified keys.

Action: Check to see that the AIX is defined correctly, and that the correct base cluster is used. If a compression package is in use, see that decompressed records are passed to DR.D.

Dispo: Processing continues, but records may be dropped from the AIX.

DOCTOR 125 ENTERED WORK= PARAMETER INVALID

Cause: The parameter entered by the 'WORK' keyword is not valid.

Action: Correct the parameter and resubmit the job if it cancels.

Dispo: Processing continues if possible.

DOCTOR 126 FILE SELECTION COMPLETE, OK TO INTERRUPT

Cause: A RESTORE=LOGICAL function with PARTIAL=SE/RE is being performed and the file selection process is complete.

Action: If you wish to skip to a new tape volume, you can now request that DRD force end-of-volume (see DOCTOR 616 message).

Dispo: Processing continues.

DOCTOR 130 DEVICE NNNN SUBCHANNEL NUMBER NOT KNOWN

Cause: During standalone restore a subchannel is specified that is invalid or not available.

Action: Restart the standalone process entering a usable number.

DOCTOR 131 SUBCHANNEL NNNN DEVICE NUMBER NOT KNOWN

Cause: During standalone restore a device is specified that is invalid or not available.

Action: Restart the standalone process entering a usable number.

DOCTOR 132 SUBCHANNEL IS DISABLED FOR DEVICE

Cause: During standalone restore a device is specified that is not enabled.

Action: Restart the standalone process entering a usable number.

DOCTOR 133 DISK COMMAND REJECT I/O FAILURE

Cause: During standalone restore a disk i/o has failed. The most likely cause is that the receiving disk is too small to contain the disk being restored.

Action: Check to see that the receiving disk is at least as large as the saved disk, and that the device type is the same. If restoring under VM, see that the virtual disk is defined at least as large as the saved disk.

DOCTOR 147 TRACK CAPACITY EXCEEDED AT CYLINDER CCCCC, HEAD HH, RECORD RRR. **RC128**

Cause: While restoring data to a CKD type disk, a block being restored will not fit in the remaining space on the track.

Action: If the backup is from a disk of the same type and track capacity, check to see that the tape has no errors and that the correct backup is on the tape by running the Dr.D FUNCTION=LTAPE. If no problems are found, contact product support.

Dispo: The job is canceled.

DOCTOR 148 FILE SAVED FROM FBA DISK IN NON-CI FORMAT CANNOT BE RESTORED TO CKD DISK- NO DATA RESTORED **RC002**

Cause: A file saved from FBA disk was in non-CI format so the logical blocksize cannot be determined for restore to a CKD type disk.

Action: None.

Dispo: The file is opened, creating a format 1 label in the VTOC, but no data is restored and processing continues. (It may be possible to restore this file type using the 'DATA TAPE' feature.)

DOCTOR 149 LAST EXTENT FOR DISK FILE 'TAPEOUT', AT DISK ADDRESS (CKD OR FBA ADDRESS HERE), LAST ADDRESS WRITTEN IS (DISK ADDRESS)

Cause: The tape file 'TAPEOUT' has been directed to disk, and at end of job, this message allows the user to see how much of the disk file has been used up.

Dispo: None, normal end of job follows.

DOCTOR 150 UNABLE TO OPEN VSAM FILE. VSAM ERR CODE=XXX **RC128**

Cause: The VSAM file could not be opened. See VSAM messages.

Action: None.

Dispo: The job is canceled.

DOCTOR 151 UNRECOVERABLE ERROR ON VSAM FILE. VSAM CODE=XXX **RC128**

Cause: See VSAM messages.

Action: None.

Dispo: The job is canceled.

DOCTOR 152 VSAM FILE IS VARIABLE, BUT VA=YES NOT SPECIFIED **RC128**

Cause: While saving a VSAM file, a logical record is encountered whose length varies from the defined size, but the backup is defined for fixed block.

Action: Put VA=YES on the save control card and rerun.

Dispo: The job is canceled.

DOCTOR 153 GETVIS UNABLE TO ALLOCATE **RC128**

Cause: Not enough memory is available for GETVIS.

Action: Allocate more memory (virtual or real) or reduce the size specified on the execute card (minimum 64K).

DOCTOR 154 SA/RE=LO SPECIFIED WITHOUT SVOL/RVOL OR SCAT/RCAT **RC128**

Cause: The SV/RV and SC/RC parameters are used to determine whether SA/RE= LOGICAL is to be driven by a VTOC or VSAM catalog.

Action: Put SVOL/RVOL or SCAT/RCAT parameters on the control card and resubmit the job.

Dispo: The job is canceled.

DOCTOR 159 NUMBER OF TRACKS TO BE SAVED- NNNNNNN, NOT EQUAL TO NUMBER SAVED- NNNNNNN **RC128**

Cause: The number of tracks to be saved in an extent backed up is not equal to the number actually passed to the output buffer.

Dispo: This is probably due to a disk I/O failure. Rerun the backup to see if the failure is consistent. Run the IBM DSF analyze function if the problem recurs.

DOCTOR 160 INVALID TIME STAMP ENTERED **RC128**

Cause: The time stamp entered (TSTAMP=) for file or member selection is either invalid or inconsistent with the IPL date, i.e. is a future date or more than five years before the IPL date.

Dispo: The job is canceled. Correct the time stamp and rerun.

DOCTOR 161 INVALID EXPIRATION PARAMETER ENTERED **RC128**

Cause: The expiration parameter entered (EX=xxx) for file selection is invalid.

Dispo: The job is canceled. Correct the parameter and rerun.

DOCTOR 170 CATALOG ERROR WHILE DEFINING DATASET 'XXXXXXXXXX **RC128**

Cause: VSAM returned an error to Dr.D while defining a VSAM cluster.

Dispo: The job is canceled. Consult the IBM manual 'VSE/VSAM' messages and codes for an explanation of the IDC message that follows this one.

DOCTOR 171 NO CATALOG INFORMATION FOR VSAM FILE XXXX CAT=XXXX **RC128**

Cause: Dr.D could find no entry in the VSAM catalog for the VSAM file. If a 'SAVE=VSAM' function is used, a JCL error is the likely cause. If a 'SAVE=LOGICAL' function is used, the file has probably been deleted by another job or task.

Action: Enter 'CANCEL' to cancel the job, or 'IGNORE' to treat the file as null (the file will be skipped during restore). If 'ERROR=IGNORE' is specified for 'SAVE=LOGICAL', the backup will proceed as though 'IGNORE' was the response to this message.

DOCTOR 172 NO LABEL INFORMATION FOR VSAM FILE XXXXXX **RC128**

Cause: Dr.D was unable to locate any label information for the VSAM file.

Action: Check JCL.

Dispo: The job is canceled.

DOCTOR 173 UNSUPPORTED VSAM OBJECT TYPE X **RC128**

Cause: Dr.D encountered a VSAM object which it did not recognize.

Action: Check JCL. If problem persists, call product support.

Dispo: The job is canceled.

DOCTOR 174 VSAM CATALOG OPEN ERROR **RC128**

Cause: VSAM returned an error when Dr.D tried to open the catalog for reading.

Action: Check the IBM manual 'VSE/VSAM' messages and codes' for an explanation of the VSAM message which precedes this one. Check JCL.

Dispo: The job is canceled.

DOCTOR 175 DATASET 'XXXXX' IN USE **RC128**

Cause: During a VSAM logical save Dr.D encountered a file open in another partition.

Dispo: Depending on the VUSE parameter.

A. The job is canceled.

B. The file is ignored and not saved.

C. The file is saved.

D. The operator is given the opportunity to respond 'IGNORE', 'CANCEL', 'NULL', 'SAVE', or Enter to retry open.

DOCTOR 176 DATASET 'XXXXXXX' EMPTY - IGNORED

Cause: The 'VNULL' parameter in a VSAM logical save was set to 'IGNORE' and Dr.D encountered an empty file.

Action: None.

Dispo: The file is ignored and processing continues.

DOCTOR 179 (DEFINING) VSAM DATASET 'XXXXXXX' (DELETING)

Cause: An information message issued during a VSAM automatic definition.

Action: None.

Dispo: Processing continues.

DOCTOR 181 NO AVAILABLE PROGRAMMER LOGICAL UNITS **RC128**

Cause: Temporary assignment(s) cannot be made due to lack of unassigned programmer logical unit(s). If 'SAVE/RESTORE= LOGICAL' (VTOC) is in operation, a contiguous set of sysno's is required, one for each disk in the disk pool specified by the 'SV/RV' parameter.

Action: Use the IBM LISTIO command to display the partition assignments. If possible, unassign enough programmer logical units to allow the assignments to be made and re-run the job.

Dispo: The job is canceled. Note: The logical units must be permanently unassigned.

DOCTOR 182 VSAM VOLUME 'XXXXXX' REQUIRED **RC128**

Cause: SA=VS has determined that this volume contains data for the cluster being saved, but the volume is not currently mounted.

Action: Mount the correct volume and press 'ENTER' or type 'CANCEL' to cancel the job.

Dispo: The SYSTEM waits for an operator response.

DOCTOR 183 NON-EXISTENT VOLSER ENTERED **RC128**

Cause: A disk VOLSER entered via SVOL= or RVOL= cannot be located.

Action: Check to see that the VOLSER(s) entered by 'SV', 'RV' are valid and that the required disk(s) is online and ready.

Dispo: The job is canceled.

DOCTOR 184 CATALOG ERROR (nn) DETECTED, TYPE C TO CANCEL; I TO SKIP THE FILE. **RC128**

Cause: THE VSAM catalog being accessed appears to be corrupted and required information cannot be found.

Action: Type C to cancel the job, or I to ignore the file and continue with the backup.

DOCTOR 186 CANADATE OR SPECIFIED VOLUME DOES NOT EXIST OR IS NOT ONLINE.

Action: Enter 'R' to RETRY after readying, 'D' to DROP volser from file candidates, or 'C' to cancel.

DOCTOR 187 WARNING - VTOC ON DISK VOLSER xxxxxx IS ABOUT TO BE MOVED TO (OR RESIZED AT) DISK ADDRESS nnnnnnnn, IS THIS CORRECT? **RC128**

Action: Check to see that the disk listed is not being used in any way that would cause the VTOC to be updated while it is being moved (this could happen anytime a file is closed). Then reply 'YES' if the move can be safely accomplished, or 'NO' to abort the move.

DOCTOR 188 NEW LOCATION SPECIFIED FOR VTOC IS INVALID OR ALREADY ALLOCATED OR NEW SIZE IS IMPROPER **RC128**

Cause: The VTOC move Function has discovered that the new VTOC location is occupied by a file, the current VTOC, the VOL1 of the disk, or is outside the range of the disk. If the size of the VTOC is being reduced, there could be labels in the VTOC that would be lost.

Action: Check a VTOC list to see if a file occupies the new location and either move the file or delete (see DELETE FUNCTION). Otherwise, choose a new location for the VTOC. If the size of the VTOC is being reduced, print the VTOC to see if labels exist in the part of the VTOC being dropped.

Dispo: Job is canceled

DOCTOR 189 EARLY EOF OCCURED WHILE PROCESSING FILE- (FILE ID) **RC002**

Cause: A control interval has been input that is not in proper CI format, possibly due to file corruption or file not closed.

Dispo: The BACKUP / COPY continues. But the file should be checked for validity.

DOCTOR 190 WARNING - MINI-DISK MMMMMM, ON DISK VOLSER VVVVVV IS ABOUT TO BE CREATED AT DISK ADDRESS AAAAAA, IS THIS CORRECT? **RC128**

Action: Check to see that the correct 'real' disk is specified and that the RBEGIN and RHALT addresses are correct. Then reply 'YES' to proceed, or 'NO' to abort the function.

DOCTOR 199 INVALID AREA=COMPACT PARAMETERS ENTERED **RC128**

Cause: The Area Compact function has found that a file extent will overlap the limits of the area being compacted.

Dispo: Check to see that the area(s) specified are correct, I.E. that no file within an area extends beyond that area.

DOCTOR 200 WORD 'EXIT' NOT IN COLUMNS 1-4

Cause: The user's first call to either GETREC or PUTREC caused it to read a control card from SYSIPT, which it did. However, the control card was either not supplied or was incorrect. A GETREC or PUTREC control card always has 'EXIT' in the first four card columns.

Action: None.

DOCTOR 201 RECFORM UNRECOGNIZED IN COLUMNS 5-6

Cause: A control card for either GETREC, GETTAPE, or PUTREC has an illegal record format code. The only valid ones are:

VS	VSAM
IS	VSAM
FB	FIXED BLOCKED
FU	FIXED UNBLOCKED
VB	VARIABLE BLOCKED
VU	VARIABLE UNBLOCKED
UN	UNDEFINED

Action: None.

DOCTOR 202 FIX/UNB-DATA SIZE WRONG (MESSAGE GENERATED BY GETTAPE, GETREC)

Cause: A record has just been encountered that is not the length specified on the control card.

Action: None.

DOCTOR 203 FIX/UNB-KEY SIZE WRONG (MESSAGE GENERATED BY GETTAPE GETREC)

Cause: A record has just been encountered whose key is not equal to the length on the control card.

Action: None.

DOCTOR 204 INVALID PHASE NAME USEREXIT=XXX...XXX

Where: XXX...XXX is the subject of the USEREXIT=KEYWORD or a save/restore data file control card.

Cause: The first sub-field of user exit is a user exit phase name. The supplied phase name does not conform to DOS rules (illegal character, or length) or it is absent. Also, are slashes (/) separating the sub-fields.

Action: None.

DOCTOR 205 INVALID TYPE EXIT USEREXIT=XXX...XXX

Where: XXXXX is the subject of the USEREXIT=KEYWORD or a save/restore data file control card.

Cause: The second subfield of USEREXIT=IS the type of exit. The only permissible characters are IO or EX.

Action: None.

DOCTOR 206 INVALID LOAD ADDRESS USEREXIT=XXX...XXX

Where: XXX...XXX is the subject of the USEREXIT=KEYWORD or a save/restore data file control card.

Cause: The third subfield of user exit is the load information. The first character can only be an 'A' or 'L'. Six characters must follow it. If the first was an 'L' the following six must be 0-9. If the first was an 'A' the following six must be 0-9 or A-F.

Action: None.

DOCTOR 207 INVALID FORMAT/SOFTWARE ERROR

Cause: The routine has lost its way, probably in GETTAPE, GETREC, PUTREC. If the user is satisfied that: (a) the correct disk file is saved, if applicable or (b) the correct save tape is read, and (c) the control card reflects reality, then Dr.D is at fault.

Action: Call product support.

DOCTOR 208 USER NOT FINISHED READING FILE

Cause: User program failed to read to end of input file before returning to Dr.D interface.

Action: Users should wait for the 'EOD' before returning.

Dispo: The job is canceled .

DOCTOR 209 USER CALLING FOR MORE RECORDS AFTER FILE END

Cause: User program requested more input records after end of file was indicated.

Action: None.

Dispo: The job is canceled.

DOCTOR 210 FIX/BLK BLOCK SIZE NOT EVENLY DIVISIBLE BY BLOCKING FACTOR

Cause: Control card in error as the block/record size.

Action: Correct control card and rerun job.

Dispo: The job is canceled.

DOCTOR 211 FIX/BLK RECORD WITH KEY ENCOUNTERED

Cause: Control card specified fix-blk but record has a key

Action: None.

Dispo: The job is canceled.

DOCTOR 213 NON-NUMERIC FIELD ON CONTROL CARD

Cause: Control card error.
Action: Correct control card and rerun job.
Dispo: The job is canceled.

DOCTOR 214 VAR/UNB TOO LARGE A RECORD READ

Cause: Logical record length exceeds block length.
Action: None.
Dispo: The job is canceled.

DOCTOR 215 VAR/UNB WRONG LENGTH KEY

Cause: Actual key length disagrees with control card.
Action: None.
Dispo: The job is canceled.

DOCTOR 216 VAR/UNB BLOCK PREFIX NOT EQUAL ACTUAL BLOCK SIZE

Cause: Actual block length not equal to prefix.
Action: None.
Dispo: The job is canceled.

DOCTOR 217 VAR/UNB RECORD PREFIX INCORRECT

Cause: Prefix agrees with block length but not the logical record length.
Action: None.

DOCTOR 221 USER PARAMETER PASSED TO SETTAPE INVALID

Cause: Invalid parameter passed to user exit routine.
Action: None.
Dispo: The job is canceled.

DOCTOR 222 PARTIAL SELECT/REJECT SPECIFIED, BUT NO SELECTION DATA ENTERED **RC128**

Action: None.
Dispo: The job is canceled.

DOCTOR 223 LOAD FAILED FOR USER EXIT PHASE- XXXXXXXX **RC128**

Action: Check to see that the correct user exit program name is specified by the USEREXIT=NF/ keyword.
Dispo: The job is canceled.

DOCTOR 250 BACKUP FILE DISK TO TAPE- BACKUP FILE WAS CREATED ON MM/DD/YY AT HH:MM:SS

Cause: The emulated 'TAPEOUT' file is being copied from disk to tape. The disk backup file was created on the date and time printed.
Action: None.
Dispo: The disk to tape function proceeds.

DOCTOR 251 SAVE=XX XXX SF=XXXXXXX/VOL=XXXXXX ID=XXX...XXX/START=X, STOP=Y

Cause: Information about each saved object is printed during copy of emulated 'TAPEOUT' file to tape.
Action: None.
Dispo: The disk to tape function proceeds.

DOCTOR 252 TOTAL NUMBER OF FILES SAVED-NNNN

Cause: The copy of the emulated 'TAPEOUT' file from disk to tape is completed.
Action: None.

DOCTOR 300 SYS001 IS NOT VALID DEVICE

Cause: During DOCTORVL operation. SYS001 is not assigned to a disk device.

Action: None.

Dispo: The job is canceled.

DOCTOR 302 FUNCTION NOT 'SAVE' OR 'RESTORE'

Cause: During DOCTORVL operation the function specified is not SAVE/RESTORE.

Action: None.

Dispo: The job is canceled.

DOCTOR 303 TAPE IS NOT OUTPUT FROM SAVE LOGICAL

Cause: The input tape to DOCTORVL is not a DOCTORVL format tape.

Action: None.

Dispo: The job is canceled.

DOCTOR 304 SA/RE LOGICAL CONTROL CARD INVALID/OMITTED

Cause: Invalid control card input.

Action: Check to see that SCAT/RCAT or SVOL/RVOL are present.

Dispo: The job is canceled.

DOCTOR 305 UNABLE TO OPEN VSAM CATALOG

Cause: Probable user JCL error. The VSAM catalog is opened as a logical VSAM file and the JCL must be specified as such.

Action: None.

Dispo: The job is canceled.

DOCTOR 306 ISAM FILE NOT SAVED DUE TO MULTI-VOL EXTENTS, FILE ID IS 'X'

Cause: An ISAM file resides on multiple volumes and 'AV' was not specified or 'AV' was specified and the missing extents were not found on the alternate volume.

Action: The operator can continue with the backup or cancel. If a pool of disks is being saved, check to see that all disks in the pool have been specified, or that the file can be excluded and the job continues.

Dispo: The message is displayed and the job continues.

DOCTOR 307 VOLUME SEQUENCE ERROR PROCESSING FILE-ffffffffffff **RC001**

Cause: During SAVE=LOGICAL (VTOC) a file has been selected for backup, but an extent is missing, and the file cannot be saved.

Action: The operator can 'EOB' the message and the job will continue or can enter 'CANCEL' to cancel the job.

Dispo: The message is displayed and the job pauses for operator input.

DOCTOR 308 RESTORE OVERFLOWS DISK OR DISK POOL **RC128**

Cause: During a reorganizing RESTORE=LOGICAL (VTOC) the current file being restored cannot fit into the available space.

Action: Increase the disk space or reduce the space required, if possible.

Dispo: The job is canceled.

DOCTOR 310 INVALID KEYWORD/KEYWORD OPERAND **RC128**

Cause: SAVE/RESTORE LOGICAL (VTOC) has detected an invalid keyword-operand combination.

Action: None.

Dispo: The job is canceled.

DOCTOR 311 DUPLICATE VOLSER ENTERED **RC128**

Cause: During SAVE/RESTORE LOGICAL (VTOC) processing, a duplicate VOLSER was entered via the SV/RV parameter.

Action: None.

Dispo: The job is canceled.

DOCTOR 312 NO OUTPUT VOLSER ENTERED **RC128**

Cause: During RESTORE LOGICAL (VTOC) processing, no output VOLSER has been entered via the 'RV' parameter.

Action: None.

Dispo: The job is canceled.

DOCTOR 313 ERROR IN FILE SEQUENCE SET **RC128**

Cause: VSAM backup has failed due to an error in the file's sequence set.

Action: None.

Dispo: The job is canceled.

DOCTOR 316 NO FILES SELECTED FOR BACKUP/RESTORE **RC128**

Cause: Save/Restore Logical (VSAM or VTOC) has selected no files.

Action: Correct and rerun job or add 'ERROR=IGNORE' parameter.

Dispo: Job is canceled

DOCTOR 317 WARNING- UNMATCHED SELECTOR(S) ENTERED **RC001**

Cause: A function using selector cards has one or more selectors that are unmatched. The unmatched selectors are printed on the report for analysis. This message will appear on the log along with the report if a backup/restore/copy function is in use. If ERROR=IGNORE is used, the log output is suppressed.

Action: If necessary, correct and rerun the job.

Dispo: Processing continues.

DOCTOR 401 Dr.D/ON-LINE INTERFACE TERMINATED

Cause: The current function completed or was terminated.

Action: The Dr.D/ON-LINE transaction terminates.

DOCTOR 402 BMS MAP FAIL OCCURED

Cause: An I/O error occurred during the sending/receiving of a Dr.D/ON-LINE BMS map.

Action: Retry Dr.D/ON-LINE transaction on another terminal. If the problem continues contact Dr.D Technical Support.

DOCTOR 403 BMS MAP WAS NOT FOUND

Cause: A Dr.D/ON-LINE BMS map is unknown to this CICS systems.

Action: Make sure the CICS installation procedures have been followed and the library containing the Dr.D/ON-LINE feature is available.

DOCTOR 404 BMS LOGIC ERROR OCCURED

Cause: A CICS error occurred during a Dr.D/ON-LINE mapping function.

Action: This is probably a CICS error. Make sure the appropriate CICS BMS support has been generated.

DOCTOR 405 PROGRAM CONTROL LINK FAILED FOR: xxxxxxxx

Cause: The program named in the message cannot be located.

Action: Make sure the program is in the PPT and the library containing the program is available to CICS.

DOCTOR 406 TERMINAL RESET NO ALLOWED HERE

Cause: A request to reset the terminal has been made.

Action: Request ignored.

DOCTOR 407 TERMINAL RESET FAILURE nnn

Cause: An error occurred during the terminal reset processing.

Action: Contact the Doctor D technical support group.

DOCTOR 408 SELECTION NOT VALID-PLEASE RETRY

Cause: An incorrect function selection was entered.

Action: The system waits for another selection to be made.

DOCTOR 409 HIGHLIGHTED FEILDS ARE INVALID

Cause: Invalid fields have been entered in the selection map.

Action: Correct the fields and enter the data.

DOCTOR 410 NO DATA ENTERED IN PANEL-PLEASE RETRY

Cause: There was no data entered in the selection panel.

Action: The systems waits until data is entered into the selection map.

DOCTOR 411 DUPLICATE FILENAMES HAVE BEEN SPECIFIED

Cause: Two or more file selection entries have the same filename.

Action: Filename must be unique. Correct the duplicate filename entries and re-enter the data.

DOCTOR 412 NO DYNAMIC USER STORAGE AVAILABLE

Cause: A request for dynamic CICS storage failed.

Action: Increase the dynamic CICS storage by increasing the partition size or run this CICS in a larger partition.

DOCTOR 413 FUNCTION IS NOT AVAILABLE

Cause: The code for the function is not available with this release.

Action: Select another function.

DOCTOR 414 ERROR DURING Dr.D CTL STMT BUILD

Cause: An internal logic error occurred during the building of the Dr.D control statements in module DRD$BDRD.

Action: A CICS transaction dump will occur with a dump code of 'Dr.D2'. Contact the Dr. D technical support group. Please have a dump available for problem analysis.

DOCTOR 415 ERROR DURING JOB STREAM BUILD

Cause: An logic error occurred during the building of the VSE job stream in module DRD$BJOB.

Action: A CICS transaction dump will occur with a dump code of 'DRD2'. Contact the Dr.D technical support group. Please have a dump available for problem analysis.

DOCTOR 416 JOB SUBMIT FAILURE, FUNCTION= nnn RC =nnn/nnn

Cause: A CICS/POWER error occurred during the submit of the Dr.D VSE job stream in module DRD$BJOB.

Action: Contact the Dr.D technical support group. Please note the function and return code values in the message. These fields provide additional information on the failure.

DOCTOR 417 JOB HAS BEEN SUBMITTED

Cause: The Dr.D Batch job stream has been submitted to power.

Action: The current Dr.D/ON-LINE function has completed.

DOCTOR 418 DUPLICATED VOLIDS HAVE BEEN SPECIFIED

Cause: A DASD volid has been specified more than once in the DASD volume copy information panel.

Action: The DASD VOL id's must be unique. Correct panel and then re-enter it.

DOCTOR 419 NO COPY INFORMATION TO PROCESS

Cause: The PF5 key has been depressed yet no copy information has been supplied.

Action: Either enter copy information on the current panel or use the PF5 key to exit this function.

DOCTOR 420 SOURCE/TARGET LIBRARIES ARE THE SAME

Cause: The filename of the target library is identical to the filename of the source filename.

Action: Dr.D does not support copying to the same library. Correct the panel or press PF3 to cancel the function.

DOCTOR 610 *WARNING* TAPE COPYING ABOUT TO BEGIN- TAPES WILL NOT BE OPENED, AND OUTPUT TAPES ARE OVERWRITTEN DESTROYING ALL DATA ON THEM, YOU ARE SET TO USE INPUT TAPE DRIVE CCUU AND OUTPUT TAPE IS THIS CORRECT?

Cause: The TAPECOPY FUNCTION is in use, and about to start.

Action: Be sure that only expendable tapes are mounted on the output tape drive(s) listed. Be sure that everyone is aware of the use of the drives and that no tapes are accidentally mounted (a sign on the drive(s) might be prudent). Then enter 'YES' to begin copying or 'NO' to cancel.

DOCTOR 612 ASSIGNMENTS INCORRECT FOR TAPE COPY FUNCTION

Cause: The tape assignments for tape copying are incorrect.

Action: Check the assignments, insuring that SYS007 is assigned to tape and if UPSI xxx1 is set that SYS006 is assigned to a different tape. Also check that SYS008 is assigned to the input tape drive.

DOCTOR 613 END OF OUTPUT TAPE REACHED BEFORE END OF INPUT TAPE- HIT ENTER TO CONTINUE ON ANOTHER TAPE OR TYPE END TO TREAT AS END OF INPUT.

Cause: While copying a tape in image mode, end of tape was reached on the output tape before it was reached on input.

Action: If you determine that all needed data was copied, type 'end', otherwise hit enter to continue on another tape.

DOCTOR 614 ASSIGNMENTS INCORRECT FOR TAPE TEST FUNCTION

Cause: The tape assignments for tape testing are incorrect.

Action: Check the assignments, insuring that SYS007 is assigned to tape and if UPSI xxx1 is set that SYS006 is assigned to a different tape.

DOCTOR 615 *WARNING* TAPE TESTING ABOUT TO BEGIN- TAPES WILL NOT BE OPENED, AND ARE OVERWRITTEN DESTROYING ALL DATA ON THEM. YOU ARE SET TO USE TAPE DRIVE IS THIS CORRECT?

Cause: The TAPETEST FUNCTION is in use, and about to start.

Action: Be sure that no non-expendable tapes are mounted on the tape drive(s) listed. Be sure that everyone involved is aware of the use of the drives and that no tapes are accidentally mounted (a sign on the drive(s) might be prudent). Then enter 'YES' to begin testing or 'NO' to cancel.

DOCTOR 616 ENTER YOUR REQUEST

Cause: Operator has entered VSE MSG command to partition.

Action: Early EOV (end of volume) can be requested for any input or output tape by entering 'EOV TAPEIN', 'EOV TAPEOUT' or 'EOV TAPETWO'. This can be useful in several situations: For backup, the operator may observe that the output tape is experiencing a high error level, reducing throughput. This allows the backup to continue on another reel. For the 'RESTORE=LOGICAL' function, the restore must begin with the reel containing the VTOC

or VSAM CATALOG contents (first backup file). After the contents info is read, a search for the selected file(s) is started, and EOV may be forced and the reel containing the file(s) mounted.

Dispo: The job will continue normally with the next reel. The user must insure that the correct reel is mounted at the appropriate time when restoring or the search may fail.

DOCTOR 617 UNRECOVERABLE ERROR HAS OCCURRED ON OUTPUT TAPE WHILE SAVING .....(OBJECT BEING SAVED)..... CAT=CCCCCC (CUU=VVVVVV) ENTER CANCEL TO ABORT, OR YES TO RECOVER BY FORCED END OF TAPE VOLUME.

Action: Enter Cancel to Abort, or Yes to recover by forced end of tape volume.

Dispo: If CANCEL is entered the job is canceled. If YES is entered the tape is positioned and forced to EOV (end of volume) allowing a correct backup to be produced in spite of the tape error. The operator should schedule the reel or cartridge for cleaning or disposal.

Note: The error message contains information on the type of file or disk that was in the process of being saved, when the unrecoverable error occurred.

DOCTOR 666 PRODUCT WILL EXPIRE in 'xxx' DAYS

Cause: The product has started a count down- when the number of days reaches zero the product will cease functioning.

Action: If you are a customer, please call product support to get a permanent password for this product. If you are a trial and are still testing the product, then you must call product support to get a password to extend the trial period.

Dispo: The job continues- the product will continue to function until the number of days is zero.

DOCTOR 992 INTERVENTION REQUIRED ON CCUU, READY DEVICE AND PRESS ENTER

Cause: A required device must be readied so that the standalone restore can continue. Ready the device and then press the enter key.

DOCTOR 993 STANDALONE RESTORE PROCESSING COMPLETED

Cause: All standalone restore processing is complete, and no more restores can be invoked without a re-ipl of the DR.D standalone supervisor.

DOCTOR 994 NEED TAPE VOLUME SEQUENCE xx, TAPE MOUNTED WAS yy

Cause: During standalone restore, a tape was mounted that was not the next tape in volume sequence order.

Action: The rejected tape (yy) is unloaded, mount the correct tape (xx).

DOCTOR 996 ENTER VOLSER OF DISK TO RESTORE

Cause: Standalone restore inquiry.

Action: Enter the VOLSER of the saved disk to be restored, or hit enter if the disk is the next on the tape.

DOCTOR 997 ENTER ALTERNATE TAPE CUU IF DESIRED

Cause: Standalone restore function inquiry.

Action: Hit enter if no alternate tape is to be used, or enter either 'CUU' (the alternate) or 'CUU,CUU' (primary, alternate) or 'CUU,' (primary, no alternate) if a drive other than the ipl'd drive is to be used for restoring. See standalone restore section for more explanation.

DOCTOR 998 STANDALONE RESTORE- MOUNT REEL 'nn'

Cause: During a standalone restore, end of volume has been reached and the next volume is needed.

Action: Mount the next tape volume. If an alternate tape drive is in use, be sure to use the right drive. Hit enter when the next tape is mounted. If the tape label sequence number does not match the 'nn', the tape is unloaded and the message is repeated.

DOCTOR 999 COMPONENT PROGRAM LOGIC ERROR XXXX

Cause: Dr.D has diagnosed an internal error.
Action: Call Dr.D software support.
Dispo: The job is canceled.

Conversion Utility Messages

The following messages are produced by the Dr.D Conversion Utility:

DRDCNVCU-01 CONVERT CONTROL STATEMENT MISSING

Cause: The mandatory Convert control statement is not the first input statement or what not passed via the EXEC parameter facility.

DRDCNVCU-02 INPUT I/O ERROR OCCURRED

Cause: A physical I/O error occurred during the reading of the input statements.

DRDCNVCU-03 INVALID CONVERSION TYPE

Cause: The specification for the CONVERT= keyword is unknown.

DRDCNVCU-04 LOAD FAILURE FOR XXXXXXXX

Cause: The phase xxxxxxxx is not resident on any of the current libraries referenced.

DRDCNVCU-05 INVALID COMMAND OPERAND

Cause: An invalid value was specified for one of the CONVERT keywords.

DRDCNVCU-06 OPEN FAILED FOR XXXXXX

Cause: An OPEN request failed for the file xxxxxx.

DRDCNVCU-07 NO INPUT RECORDS PROCESSED

Cause: No control statements were found in the input file.

DRDCNVCU-08 CONVERSION FAILED, RC=nnn

Cause: Based upon the RC=nnn code:

001 Invalid calling function
002 Insufficient virtual storage
003 Conversion module failed
004 No control statements to convert
005 // Exec statement conversion failure
006 Unknown conversion error code
007 Print I/O error occurred

DRDCNVCU-10 INVALID CONVERT STATEMENT KEYWORD

Cause: An invalid keyword was specified in the Convert control statement.

DRDCNVCU-11 INPUT RECORDS READ: NN,NNN

Cause: All of the input records have been read.

DRDCNVCU-12 OUTPUT RECORDS WRITTEN: NN,NNN

Cause: The conversion completed successfully and a count of the records written to the punch file is being displayed.

DRDCNVRS-02 INVALID STATEMENT FORMAT

Cause: The format of this statement is not valid.

DRDCNVRS-03 INVALID COMMAND NAME - ccc

Cause: The command (ccc) is not valid for this type of conversion.

DRDCNVRS-04 INVALID SPEC, FIELD='XXXXXXXXXXXXXXXX'

Cause: An invalid specification was encountered in this control statement.

DRDCNVRS-05 CONVERT LOGIC ERROR NNN

Cause: A logic error occurred during the conversion. Please contact BIM if this message appears.

DRDCNVRS-06 UNSUPPORTED CONVERSION OPTION='xxxxxxxxxxxxxxxx'

Cause: The conversion system cannot determine how to convert the specified statement or statement option.

Appendix XI - Sample Reports

```
1V654 DR.D LIST VTOC                                    DOSRES SYSWK1                                          12/11/1998  PAGE 001
                                                                                                                       01:46 PM
 TIME STAMP SELECTOR:NONE                    --FILE--                        LAST    BLK OR ------------EXTENT INFORMATION-----------
 ------------FILE IDENTIFICATION----------- TYPE SEC CREATED   EXPIRES    ACCESSED  CISIZE VOLSER   START     END   START    SIZE SEQ

 DOS.LABEL.FILE.FF230FFF7490.AREA1            DA  N  1/13/1998  1999/366            421376 DOSRES    59136   59455  59136    320   1
 DOS.PAGING.FILE.FF230FFF7490                 DA  N  1/14/1998  1999/366            421376 DOSRES   125568  142463 125568  16896   1
                                                                                            DOSRES   315072  331455 315072  16384   2
                                                                                            DOSRES   331456  446143 331456 114688   3
                                                                                            SYSWK1   376704  393599 376704  16896   4
                                                                                                                FILE TOTAL  164864
 ICCF.LIBRARY                                 DA  N  1/13/1998  1999/366              2048 SYSWK1   306688  376703 306688  70016   1
 INFO.ANALYSIS.DUMP.MGNT.FILE                 SD  N  1/13/1998  1/13/1998              512 SYSWK1   478912  479103 478912    192   1
 INFO.ANALYSIS.EXT.RTNS.FILE                  SD  N  1/13/1998  1999/366               512 SYSWK1   479104  479167 479104     64   1
 SYS.NEW.RES                                  DA  N  8/17/1998  8/24/1998              512 SYSWK1        2   59135      2  59134   1
 VSE.DUMP.LIBRARY                             SD  N  1/13/1998  1999/366               512 SYSWK1   196672  233727 196672  37056   1
 VSE.HARDCOPY.FILE.DEVTEPIC.ESA23             DA  N  4/06/1998  1999/366               512 SYSWK1   485168  488167 485168   3000   1
 VSE.HARDCOPY.FILE.DEVTEPI2.ESA23             DA  N  4/06/1998  1999/366               512 SYSWK1   482168  485167 482168   3000   1
 VSE.HARDCOPY.FILE.DEVTEPI3.ESA23             DA  N  4/30/1998  1999/366               512 SYSWK1   479168  482167 479168   3000   1
 VSE.HARDCOPY.FILE.TECHEPIC.VSE23             DA  N  5/12/1998 12/31/1999              512 SYSWK1   754048  759103 754048   5056   1
 VSE.HARDCOPY.FILE.VSE22.111111               DA  N  4/06/1998  1999/366               512 SYSWK1   467776  472831 467776   5056   1
 VSE.POWER.ACCOUNT.FILE                       DA  N  1/13/1998  1999/366                 0 SYSWK1   465600  467647 465600   2048   1
 VSE.POWER.DATA.FILE                          DA  N  1/13/1998  1999/366                 0 SYSWK1   393600  465599 393600  72000   1
 VSE.POWER.QUEUE.FILE                         DA  N  1/13/1998  1999/366                 0 DOSRES    59456   59839  59456    384   1
 VSE.RECORDER.FILE                            DA  N  5/12/1998  1999/366               512 SYSWK1   472832  473855 472832   1024   1
 VSE.SYSRES.LIBRARY                           DA  N  8/24/1998  1999/366               512 DOSRES       2   59135      2  59134   1
 VSE.SYSTEM.HISTORY.FILE                      DA  N  8/24/1998  1999/366              2048 DOSRES   310016  315071 310016   5056   1
 VSESP.JOB.MANAGER.FILE                       SD  N  8/17/1998  1999/366              1024 SYSWK1   467648  467775 467648    128   1
 VTAM.TRACE.FILE                              SD  N  9/17/1998  1999/366              2560 SYSWK1   473856  476855 473856   3000   1
 WORK.HIST.FILE                               DA  N  8/24/1998  8/31/1998             2048 SYSWK1    59136   64191  59136   5056   1
 Z9999992.VSAMDSPC.TAFDB902.TB9BE89D          VSAM   1/13/1998  1999/366                 0 DOSRES    63104  125503  63104  62400   1
 Z9999992.VSAMDSPC.TAFDB902.TCA5F0CE          VSAM   1/13/1998  1999/366                 0 SYSWK1   236608  306623 236608  70016   1
 Z9999992.VSAMDSPC.TAFDB902.T7B50BCE          VSAM   1/13/1998  1999/366                 0 DOSRES   142464  297983 142464 155520   1
 Z9999992.VSAMDSPC.TAFDB902.T834F4BA          VSAM   1/13/1998  1999/366                 0 SYSWK1    64192  196671  64192 132480   1
 Z9999992.VSAMDSPC.TAFDB902.T9082927          VSAM   1/13/1998  1999/366                 0 SYSWK1   489088  754047 489088 264960   1
 Z9999994.VSAMDSPC.TAFDB902.TA4631FF          VSAM   1/13/1998  1999/366                 0 SYSWK1   233728  236607 233728   2880   1
 Z9999996.VSAMDSPC.TAFDB902.T65B6CD8          VSAM   1/13/1998  1999/366                 0 DOSRES    60224   63103  60224   2880   1

                                                                                            FILES LISTED=        28        1189724
```

```
1V654 DR.D LIST VTOC                                    DOSRES SYSWK1                                          12/11/1998  PAGE 002
                                                                                                                       01:46 PM
 SPACE MAPS          ---DISK---    ------VTOC------     DISK        GAP  ---GAP START---   ----GAP END----
                     VOLSER CUU     START       END     SIZE       SIZE  TRK/BLK  CYL-HD   TRK/BLK  CYL-HD

                     DOSRES 140    921570    921599   921600        384    59840             60223
                                                                     64   125504            125567
                                                                  12032   297984            310015
                                                                 475426   446144            921569

                                TOTAL FREE SPACE ON  DOSRES    487906       52%

                     SYSWK1 141    921722    921751   921752         64   306624            306687
                                                                   2056   476856            478911
                                                                    920   488168            489087
                                                                 162618   759104            921721

                                TOTAL FREE SPACE ON  SYSWK1    165658       17%

                                TOTAL FREE SPACE ON FBA DISK   653564       35%
```

```
1V654 DR.D LISTCAT                        LISTING OF VSAM CATALOG-VSESP.USER.CATALOG                          12/11/1998  PAGE 001
 TIME STAMP SELECTOR:NONE                                                                                      01:46 PM
 SHR --KEY--- ---LRECL---        ---ALLOCATIONS--- -FREE-SPACE- SPLITS    --------LOGICAL RECORDS--------- ----EXTENT INFORMATION----
 OPT LEN  RKP AVERAGE MAX CISIZE  PRIMARY SECONDRY CI CA   KBYT CI  CA    TOTAL DELETD INSRTD UPDTED  READ VOLSER   START SIZE %USD #

 CICS.AUTO.STATS.A (MSAM,CREATED-1998/013 EXPIRES-1998/013 TIME STAMP-1998/013 00:00AM)
 1,3   0    0   304   304   2048     64B0     64B0  0  0     32   0  0       0      0      0      0      0 DOSRES  78208     64     1
 REUSE SPEED                                              99%        EXCPS=0           BUFFSPACE=   4096

 CICS.AUTO.STATS.B (MSAM,CREATED-1998/013 EXPIRES-1998/013 TIME STAMP-1998/013 00:00AM)
 1,3   0    0   304   304   2048     64B0     64B0  0  0     32   0  0       0      0      0      0      0 DOSRES  78272     64     1
 REUSE SPEED                                              99%        EXCPS=0           BUFFSPACE=   4096

 CICS.CSD (KSDS,CREATED-1998/013 TIME STAMP-1998/344 10:07PM)
 2,3  22    0   100   500   2048   1024B0    256B0  0  0      0 121  2    1760   1582   2982     24  75678 DOSRES  79744   1024 99  1
                           1024I    64B0     64B0           %    EXCPS=12080           BUFFSPACE=   5120 DOSRES  80768     64 16  1
                                                                                       INDEX LEVELS= 002

 CICS.DUMPA (MSAM,CREATED-1998/013 EXPIRES-1998/013 TIME STAMP-1998/229 03:54PM)
 1,3   0    0  7161  7161   7168   4800B0       B0  0  0   2444   0  0       0      0      0      0      0 DOSRES  80832   4800     1
 REUSE SPEED                                              99%        EXCPS=0           BUFFSPACE=  14336
                                                                                                          SYSWK1   NO ALLOCATION

 CICS.DUMPB (MSAM,CREATED-1998/014 EXPIRES-1998/014 TIME STAMP-1998/229 04:33PM)
 1,3   0    0  7161  7161   7168   1920B0       B0  0  0    974   0  0       0      0      0      0      0 SYSWK1 236736   1920  1  1
 REUSE SPEED                                              99%        EXCPS=0           BUFFSPACE=  14336
                                                                                                          DOSRES   NO ALLOCATION

 CICS.RSD (KSDS,CREATED-1998/013 TIME STAMP-1998/229 04:33PM)
 2,3  22    0  2000  2000   2048   1344B0    448B0 20 20    229  56  1    1299    604   1902  35948  72901 DOSRES  78336   1344 67  1
                           1536I    64B0     64B0           33%    EXCPS=43380           BUFFSPACE=   5632 DOSRES  79680     64 14  1
                                                                                       INDEX LEVELS= 002

 CICS.TD.INTRA (ESDS,CREATED-1998/013 TIME STAMP-1998/229 04:33PM)
 2,3   0    0  4089  4089   4096    832B0    832B0  0  0      0   0  0       0      0      0      0      0 DOSRES  76544    832 99  1
                                                          %      EXCPS=138           BUFFSPACE=   8192

 DEFAULT.MODEL.ESDS.SAM (MSAM,CREATED-1998/013 TIME STAMP-1998/013 00:00AM)
 1,3   0    0  2000  2000   2048    111B0    111B0  0  0      0   0  0       0      0      0      0      0 DOSRES   NO ALLOCATION
 REUSE FIXBLKSPEED       NOALOC                           %        EXCPS=0           BUFFSPACE=   4096 SYSWK1   NO ALLOCATION

 DFHTEMP (ESDS,CREATED-1998/013 TIME STAMP-1998/229 04:33PM)
 2,3   0    0  4089  4089   4096    832B0    832B0  0  0      0   0  0       0      0      0      0      0 DOSRES  77376    832 99  1
                                                          %      EXCPS=192           BUFFSPACE=   8192

 DRD.MASTER.WORK (MSAM,CREATED-1998/261 EXPIRES-1991/033 TIME STAMP-1998/261 04:17PM)
 1,3   0    0 16000 16000  32768   6720B0   1920B0  0  0    983   0  0       0      0      0      0      0 DOSRES  85696   6720 82  1
 REUSE SPEED                                              18%        EXCPS=0           BUFFSPACE=  65536 DOSRES  92416   1920     2
                                                                                                          DOSRES  94336   1920     3
                                                                                                          SYSWK1   NO ALLOCATION

 VSAM.COMPRESS.CONTROL (KSDS,CREATED-1998/013 EXPIRES-1999/366 TIME STAMP-1998/261 04:17PM)
 4,4  44    0   128   500   2048     64B0     64B0  0  0      0   0  0       0      1      0      0      1 SYSWK1 236608     64 99  1
                            512I    64B0     64B0           %       EXCPS=20           BUFFSPACE=   4608 SYSWK1 236672     64  2  1
                                                                                       INDEX LEVELS= 001

 VSE.CONTROL.FILE (KSDS,CREATED-1998/013 EXPIRES-1999/366 TIME STAMP-1998/229 04:33PM)
 4,3  12    0   100  1000   4096     64B0     64B0  0  0      0  11  1     199      1    191      1    723 DOSRES  71424     64 99  1
                            512I    64B0     64B0           %     EXCPS=1261           BUFFSPACE=   8704 DOSRES  85632     64     2
```

```
1V654 DR.D LISTCAT                        LISTING OF VSAM CATALOG-VSESP.USER.CATALOG                          12/11/1998  PAGE 002
 TIME STAMP SELECTOR:NONE                                                                                      01:46 PM
 SHR --KEY--- ---LRECL---        ---ALLOCATIONS--- -FREE-SPACE- SPLITS    --------LOGICAL RECORDS--------- ----EXTENT INFORMATION----
 OPT LEN  RKP AVERAGE MAX CISIZE  PRIMARY SECONDRY CI CA   KBYT CI  CA    TOTAL DELETD INSRTD UPDTED  READ VOLSER   START SIZE %USD #

                                                                                    INDEX LEVELS=   2 DOSRES  71488     64  5  1

 VSE.MESSAGE.ROUTING.FILE (KSDS,CREATED-1998/013 EXPIRES-1999/366 TIME STAMP-1998/229 04:33PM)
 2,3  16    0   500  4080   4096    256B0    256B0 15  7      0   0  0       1      0      0      0      0 DOSRES  76224    256 99  1
                           512I    64B0     64B0           %       EXCPS=27          BUFFSPACE=   8704 DOSRES  76480     64  2  1
                                                                                    INDEX LEVELS= 001

 VSE.ONLINE.PROB.DET.FILE (KSDS,CREATED-1998/013 TIME STAMP-1998/229 04:33PM)
 2,3   2    0  4000  4089   4096   2496B0    832B0  0  0    425   0  0     202      0      0     58   5974 DOSRES  68864   2496 67  1
                          1024I    64B0     64B0          33%     EXCPS=6182          BUFFSPACE=   9216 DOSRES  71360     64  9  1
                                                                                    INDEX LEVELS= 002

 VSE.PRIMARY.LIBRARY (MSAM,CREATED-1998/013 TIME STAMP-1998/013 10:54PM)
 3,3   0    0  4089  4089   4096   5760B0  17280B0  0  0      0   0  0       0      0      0      0      0 DOSRES  63104   5760 99  1
 SPEED                                                  %        EXCPS=0           BUFFSPACE=   8192
                                                                                                 SYSWK1   NO ALLOCATION

 VSE.TEXT.REPSTORY.FILE (KSDS,CREATED-1998/013 EXPIRES-1999/366 TIME STAMP-1998/229 04:33PM)
 2,3  12    0   500  4080   4096   4608B0    256B0  0  0    917   2  2    2712      0      0   2709   2919 DOSRES  71552   4608 61  1
                           512I    64B0     64B0          39%     EXCPS=3288          BUFFSPACE=   8704 DOSRES  76160     64 19  1
                                                                                    INDEX LEVELS= 002

                                                                          FILES/OBJECTS LISTED-      16               35200
```

```
1V654 DR.D LISTCAT       ***SPACE MAP***    LISTING OF VSAM CATALOG-VSESP.USER.CATALOG                          12/11/1998  PAGE 003
 TIME STAMP SELECTOR:NONE                                                                                      01:46 PM
  DISK       EXTENT  ---TRACKS OR BLOCKS---  ------------------------------------G A P   M A P-------------------------------------
 VOLSER       BEGIN     TOTAL    AVAILABLE                (LISTING OF GAP SIZE X NUMBER OF OCCURENCES OF THAT SIZE)

 DOSRES      63104     62400     29248 46%   29248X1
 DISK TOTAL            62400     29248 46%

 SYSWK1     233728      2880         0  0%   67968X1
           236608     70016     67968 97%
 DISK TOTAL            72896     67968 93%

 CATALOG TOTAL        135296     97216 71%
```

```
1DR.D V654                                        LIBRARY DIRECTORY LISTING FOR                                  12/11/98  PAGE 001
 TIME STAMP SELECTOR:NONE                          BIM.LIBRARY.CURRENT.VERSION                                    01:46 PM
 LIBRARY SIZE (BLOCKS)   40320   20636(51.1%) FREE
 -----MEMBER----- ---------DATE-TIME-----------  --BLOCKS--        -----MEMBER----- ---------DATE-TIME-----------  --BLOCKS--
   NAME    TYPE   ...CREATED.... ...CHANGED....  TOTAL CNTG  SIZE    NAME    TYPE   ...CREATED.... ...CHANGED....  TOTAL CNTG  SIZE

 **************BIM.LIBRARY.CURRENT.VERSION.DRD******************  DOCTOR45 PHASE   98/08/13  9:59 98/08/13 10:01     2     2   1920
 DOCTOR   PHASE   98/08/13  9:59 98/09/18  8:57    71    71  69888  DOCTOR46 PHASE   98/08/13  9:59 98/08/13 10:01     4     4   2980
 DOCTORCN PHASE   98/08/13  9:59 98/08/13 10:00     7     7   6240  DOCTOR47 PHASE   98/08/13  9:59 98/08/13 10:01    13    13  11981
 DOCTORMS PHASE   98/08/13  9:59 98/08/13 10:00    10    10   8956  DOCTOR48 PHASE   98/08/13  9:59 98/08/13 10:01    13    13  12464
 DOCTORPC PHASE   98/08/13  9:59 98/08/13 10:00    47    47  46096  DOCTOR49 PHASE   98/08/13  9:59 98/08/13 10:01     5     5   4528
 DOCTORSF PHASE   98/08/13  9:59 98/08/13 10:00     2     2   1966  DOCTOR50 PHASE   98/08/13  9:59 98/08/13 10:01    14    14  13495
 DOCTORTC PHASE S 98/08/13  9:59 98/08/13 10:00     2     2   1224  DOCTOR51 PHASE   98/08/13  9:59 98/08/13 10:01     8     8   7604
 DOCTORUP PHASE   98/08/13  9:59 98/08/13 10:00    73    73  71925  DOCTOR52 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2859
 DOCTORVL PHASE   98/08/13  9:59 98/08/13 10:00    33    33  32294  DOCTOR53 PHASE   98/08/13  9:59 98/08/13 10:01     9     9   8168
 DOCTORWW PHASE   98/08/13  9:59 98/08/13 10:00     1     1    152  DOCTOR54 PHASE   98/08/13  9:59 98/08/13 10:01     5     5   4499
 DOCTORZZ PHASE   98/08/13  9:59 98/08/13 10:00     5     5   4388  DOCTOR55 PHASE   98/08/13  9:59 98/08/13 10:01     2     2   1976
 DOCTOR00 PHASE   98/08/13  9:59 98/08/14  7:45    71    71  69888  DOCTOR56 PHASE   98/08/13  9:59 98/08/13 10:01     2     2   1504
 DOCTOR01 PHASE   98/08/13  9:59 98/08/13 10:00     3     3   2312  DOCTOR57 PHASE   98/08/13  9:59 98/08/13 10:01     7     7   6482
 DOCTOR02 PHASE   98/08/13  9:59 98/08/13 10:01    20    20  19140  DOCTOR58 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2288
 DOCTOR03 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2952  DOCTOR59 PHASE   98/08/13  9:59 98/08/13 10:01    69    69  68012
 DOCTOR04 PHASE   98/08/13  9:59 98/08/13 10:01     1     1    604  DOCTOR60 PHASE   98/08/13  9:59 98/08/13 10:01     1     1    712
 DOCTOR05 PHASE   98/08/13  9:59 98/08/13 10:01     2     2   1320  DOCTOR61 PHASE   98/08/13  9:59 98/08/13 10:01     1     1    672
 DOCTOR06 PHASE   98/08/13  9:59 98/08/13 10:01     9     9   8776  DOCTOR62 PHASE   98/08/13  9:59 98/08/13 10:01     9     9   8168
 DOCTOR07 PHASE   98/08/13  9:59 98/08/13 10:01     2     2   1922  DOCTOR63 PHASE   98/08/13  9:59 98/08/13 10:01    16    16  15668
 DOCTOR08 PHASE   98/08/13  9:59 98/08/13 10:01    20    20  19140  DOCTOR64 PHASE   98/08/13  9:59 98/08/13 10:01     8     8   7792
 DOCTOR09 PHASE   98/08/13  9:59 98/08/13 10:01     4     4   3872  DOCTOR65 PHASE   98/08/13  9:59 98/08/13 10:01     1     1    870
 DOCTOR10 PHASE   98/08/13  9:59 98/08/13 10:01     5     5   4568  DOCTOR66 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2593
 DOCTOR11 PHASE   98/08/13  9:59 98/08/13 10:01     1     1    760  DOCTOR67 PHASE   98/08/13  9:59 98/08/13 10:01     7     7   6288
 DOCTOR12 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2537  DOCTOR68 PHASE   98/08/13  9:59 98/08/13 10:01     2     2   1076
 DOCTOR13 PHASE   98/08/13  9:59 98/08/13 10:01     8     8   7550  DOCTOR69 PHASE   98/08/13  9:59 98/08/13 10:01     9     9   8101
 DOCTOR14 PHASE   98/08/13  9:59 98/08/13 10:01    27    27  25780  DOCTOR70 PHASE   98/08/13  9:59 98/08/13 10:01     2     2   1437
 DOCTOR15 PHASE   98/08/13  9:59 98/08/14  7:45    17    17  16384  DOCTOR71 PHASE   98/08/13  9:59 98/08/13 10:01    13    13  12214
 DOCTOR16 PHASE   98/08/13  9:59 98/08/14  7:45    18    18  17653  DOCTOR72 PHASE   98/08/13  9:59 98/08/13 10:01     7     7   6828
 DOCTOR17 PHASE   98/08/13  9:59 98/08/13 10:01    10    10   9588  DOCTOR73 PHASE   98/08/13  9:59 98/08/13 10:01     5     5   4744
 DOCTOR18 PHASE   98/08/13  9:59 98/08/13 10:01     7     7   6316  DOCTOR74 PHASE   98/08/13  9:59 98/08/13 10:01    10    10   9456
 DOCTOR19 PHASE   98/08/13  9:59 98/08/13 10:01     7     7   6672  DOCTOR75 PHASE   98/08/13  9:59 98/08/13 10:01     4     4   3808
 DOCTOR20 PHASE   98/08/13  9:59 98/08/13 10:01     1     1     48  DOCTOR76 PHASE   98/08/13  9:59 98/08/13 10:01     5     5   4760
 DOCTOR21 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2024  DOCTOR78 PHASE   98/08/13  9:59 98/08/13 10:01     8     8   7444
 DOCTOR22 PHASE   98/08/13  9:59 98/08/13 10:01     5     5   4000  DOCTOR80 PHASE   98/08/13  9:59 98/08/13 10:01    13    13  12056
 DOCTOR23 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2576  DOCTOR81 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2672
 DOCTOR24 PHASE   98/08/13  9:59 98/08/13 10:01     1     1    478  DOCTOR82 PHASE   98/08/13  9:59 98/08/13 10:01     7     7   6064
 DOCTOR25 PHASE   98/08/13  9:59 98/08/13 10:01     9     9   8696  DOCTOR85 PHASE   98/08/13  9:59 98/08/13 10:01     8     8   7504
 DOCTOR26 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2340  DOCTOR86 PHASE   98/08/13  9:59 98/08/13 10:01    10    10   9740
 DOCTOR27 PHASE   98/08/13  9:59 98/08/13 10:01     5     5   4094  DOCTOR87 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2869
 DOCTOR29 PHASE   98/08/13  9:59 98/08/13 10:01     4     4   3271  DOCTOR88 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2395
 DOCTOR31 PHASE   98/08/13  9:59 98/08/13 10:01     4     4   3948  DOCTOR93 PHASE   98/08/13  9:59 98/08/13 10:01     4     4   3272
 DOCTOR32 PHASE   98/08/13  9:59 98/08/13 10:01     8     8   7589  DOCTOR97 PHASE   98/08/13  9:59 98/08/13 10:01     6     6   5150
 DOCTOR33 PHASE   98/08/13  9:59 98/08/13 10:01     6     6   5183  DOCTOR98 PHASE   98/08/13  9:59 98/08/13 10:01     5     5   4726
 DOCTOR34 PHASE   98/08/13  9:59 98/08/13 10:01    13    13  11981     TYPE TOTAL           MEMBERS=     96         924
 DOCTOR35 PHASE   98/08/13  9:59 98/08/13 10:01     1     1    464     SUB-LIBRARY:DRD      MEMBERS=     96         924
 DOCTOR36 PHASE   98/08/13  9:59 98/08/13 10:01     2     2   1152
 DOCTOR37 PHASE   98/08/13  9:59 98/08/13 10:01     2     2   1680     REPORT TOTAL         MEMBERS=     96         924
 DOCTOR38 PHASE   98/08/13  9:59 98/08/13 10:01     6     6   5127
 DOCTOR39 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2447
 DOCTOR40 PHASE   98/08/13  9:59 98/08/13 10:01     2     2   1104
 DOCTOR41 PHASE   98/08/13  9:59 98/08/13 10:01     3     3   2751
 DOCTOR42 PHASE   98/08/13  9:59 98/08/13 10:01     5     5   4332
 DOCTOR43 PHASE   98/08/13  9:59 98/08/13 10:01     9     9   8040
 DOCTOR44 PHASE   98/08/13  9:59 98/08/13 10:01     5     5   4660
```

```
1DR.D V654                                    LIBRARY DIRECTORY LISTING FOR                                    12/11/98  PAGE 002
 TIME STAMP SELECTOR:NONE                      BIM.LIBRARY.CURRENT.VERSION                                      01:46 PM
 LIBRARY SIZE (BLOCKS)   40320   20636(51.1%) FREE
                                              SUB     MEMBER        MEMBER     BLOCK   LIBRARY
                         **LIBRARY RECAP**   LIBRARY   TYPE         COUNT     COUNT    SHARE

                                             CONFIG    PHASE            4         9       .0%
                                             CONFIG    *TOTAL*          4         9       .0%

                                             CURRPTF   A               27        72       .1%
                                             CURRPTF   C               10        71       .1%
                                             CURRPTF   E                9        43       .1%
                                             CURRPTF   OBJ             10        73       .1%
                                             CURRPTF   PHASE          153      5125     12.7%
                                             CURRPTF   PROC             6      2724      6.7%
                                             CURRPTF   SAMPJOB         185       610      1.5%
                                             CURRPTF   Z               13       472      1.1%
                                             CURRPTF   *TOTAL*        413      9190     22.7%

                                             DRD       A               4        14       .0%
                                             DRD       PHASE         214      2314      5.7%
                                             DRD       *TOTAL*       218      2328      5.7%

                                             DRDOBJ    OBJ           569      6719     16.6%
                                             DRDOBJ    PHASE           1        71       .1%
                                             DRDOBJ    *TOTAL*       570      6790     16.8%

                                             DRT       A              35       122       .3%
                                             DRT       C               5        19       .0%
                                             DRT       E              26        90       .2%
                                             DRT       OBJ            21        35       .0%
                                             DRT       PHASE          82       778      1.9%
                                             DRT       *TOTAL*       169      1044      2.5%

                                             LIBRARY TOTAL          1374     19361     48.0%

                                             LIBRARY OVERHEAD BLOCKS            323       .8%
```

```
1DR.D V654                                    LIBRARY DIRECTORY LISTING FOR                                    12/11/98  PAGE 003
 TIME STAMP SELECTOR:NONE                      BIM.LIBRARY.CURRENT.VERSION                                      01:46 PM
 LIBRARY SIZE (BLOCKS)   40320   20636(51.1%) FREE
                                                     MEMBER        MEMBER     BLOCK   LIBRARY
                         **LIBRARY RECAP**            TYPE         COUNT     COUNT    SHARE

                                                     A              66       208       .5%
                                                     C              15        90       .2%
                                                     E              35       133       .3%
                                                     OBJ           600      6827     16.9%
                                                     PHASE         454      8297     20.5%
                                                     PROC            6      2724      6.7%
                                                     SAMPJOB       185       610      1.5%
                                                     Z              13       472      1.1%
                                             LIBRARY TOTAL        1374     19361     48.0%

                                             LIBRARY OVERHEAD BLOCKS            323       .8%

                                             LIBRARY EFFICIENCY RATING           92
```

Index

Index

Index

Index