A8L

1HD7

North American Response Centers

HP 3000 APPLICATION NOTE #24

PRIVATE VOLUMES

HEWLETT PACKARD

March 15, 1987
Document P/N 5958-5824R2712

HP 3000 APPLICATION NOTES *are published by the North American Response Centers twice a month and distributed with the Software Status Bulletin. These notes address topics, where the volume of calls received at the Centers indicates a need for addition to or consolidation of information available through HP support services. You may obtain previous notes (single copies only, please) by returning the attached Reader Comment Sheet listing their numbers.*

Note #	*Published*	*Topic*
1	*2/21/85*	*HP 3000 Printer Configuration Guide (superseded by note #4)*
2	*10/15/85*	*Terminal Types for HP 3000 HPIB Computers (superseded by note #13)*
3	*4/01/86*	*HP 3000 Plotter Configuration Guide*
4	*4/15/86*	*HP 3000 Printer Configuration Guide - Revised*
5	*5/01/86*	*MPE System Logfile Record Formats*
6	*5/15/86*	*HP 3000 Stack Operation*
7	*6/01/86*	*COBOL II/3000 Programs: Tracing Illegal Data*
8	*6/15/86*	*KSAM Topics: COBOL's Index I/O; File Data Integrity*
9	*7/01/86*	*Port Failures, Terminal Hangs, TERMDSM*
10	*7/15/86*	*Serial Printers - Configuration, Cabling, Muxes*
11	*8/01/86*	*System Configuration or System Table Related Errors*
12	*8/15/86*	*Pascal/3000 - Using Dynamic Variables*
13	*9/01/86*	*Terminal Types for HP 3000 HPIB Computers - Revised*
14	*9/15/86*	*Laser Printers - A Software and Hardware Overview*
15	*10/01/86*	*FORTRAN Language Considerations - A Guide to Common Problems*
16	*10/15/86*	*IMAGE: Updating to TurboImage & Improving Data Base Loads*
17	*11/01/86*	*Optimizing VPLUS Utilization*
18	*11/15/86*	*The Case of the Suspect Track for 792X Disc Drives*
19	*12/01/86*	*Stack Overflows: Causes & Cures for COBOL II Programs*
20	*1/01/87*	*Output Spooling*
21	*1/15/87*	*COBOLII and MPE Intrinsics*
22	*2/15/87*	*Asynchronous Modems*
23	*3/01/87*	*VFC Files*

NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

This document contains proprietary information which is protected copyright. All rights are reserved. Permission to copy all or part of this document is granted provided that the copies are not made or distributed for direct commercial advantage; that this copyright notice, and the title of the publication and its date appear; and that notice is given that copying is by permission of Hewlett-Packard Company. To copy otherwise, or to republish, requires prior written consent of Hewlett-Packard Company.

Copyright © 1987 by HEWLETT-PACKARD COMPANY

PRIVATE VOLUMES

MPE's Private Volumes facility is one of the most useful but misunderstood features of the HP 3000. This application note will attempt to "de-mystify" the subject of Private Volumes by describing how PVs are different from "regular" or system discs and why at times they may be preferable. It will also provide a step-by-step procedure for implementation.

DISC DOMAINS

MPE classifies discs in two ways: system or non-system. These classifications are called *domains*, and that term will be used throughout this paper. The system domain refers to all discs which are configured into the Volume Table, while the non-system domain refers to all discs configured as *foreign, serial, or private volume* (those *not* in the Volume Table).

System discs are the ones used day in and day out on most standard configurations. The best example of a system disc is *the* system disc which is always logical device one (ldev 1) and which is the only disc which *must* be in the system domain. The HP 3000 can have up to 24 system-domain discs.

Of the non-system types of disc, FOREIGN discs allow access to data which is *not* stored in standard HP 3000 MPE format. Foreign discs are also used by DISCCOPY, an IBM data transfer utility, and by COPYCAT. SERIAL discs emulate a magnetic tape drive and can be used as an alternative storage medium for :SYSDUMP, :STORE, and :RESTORE activities. PRIVATE VOLUMES are user definable volume sets (and the topic of this note).

When the system is started, MPE requires that each SYSTEM disc be defined in the system volume table, physically mounted on the system, and accessible (i.e. the drive on which the disc is mounted is UP, free of hardware problems, etc.). *If any of these conditions are not true, the system will not come up.*

The main difference between system and non-system discs is that non-system discs can be mounted and dismounted while the system is running and without interrupting normal system activity. A PRIVATE VOLUME CAN BE DOWN WITHOUT THE SYSTEM BEING DOWN TOO! If ANY system domain disc is down, the entire HP 3000 is down.

To appreciate one ramification of private volumes consider a disc failure which forces a reload. On a 3000 with only system domain discs, all drives must be "up" before the system can come up. Also, because the MPE file system allocates files to the system disc domain on whatever device it can find space (as opposed to one specific device) all discs must be reloaded or all files for a given application must be restored, before the application can run. With private volumes in the same situation only the system-domain volumes need be available in order for the system to come up. If files for specific applications are on private volume sets, only those sets need be up for the application to run. Hardware, data or software recovery can continue while your application on the PVs is running.

ADVANTAGES AND DISADVANTAGES

The primary advantage to using private volumes is increased overall system availability. There are other advantages too, and they include:

- Additional disc storage at a low cost.

In situations where all data/applications on the system do not need to be online at all times you can store your applications on private volume sets and use them only as needed. The cost of additional disc packs is considerably less than disc drives.

- File transportability.

  A private volume (and all of its data) can easily be moved from one HP 3000 to another.

- Additional level of system security.

  To secure files on a private volume a drive can be turned off or the pack can be removed. Alternatively, you can require that the operator reply to each request for access to the volume. In this way, you can control access through the need for operator intervention. Additionally, access to private volumes requires the special MPE capability UV, use volume.

- More selective and faster backup/reload.

  You can exclude private volumes from :STORE and :SYSDUMP operations by taking the volume(s) offline.

Of course, just as there is no free lunch in life, using private volumes is not without a price. Most of private volumes' disadvantages fall into the "inconvenience" category. These include:

- Additional account structure maintenance.

  This includes using the "VS=" parameter on :NEW..., :ALT..., and :PURGE... commands for accounts and groups and adding UV and CV capabilities.

- Additional I/O required during FOPEN.

  When a file on a private volume is initially FOPENed the System Directory (on ldev 1) and the private volume directory (on the master volume of the private volume set) are searched and "binding" (which is the setting of a pointer from the system directory to the PV file) occurs. This additional overhead occurs only on the first FOPEN to a file. However, once a file is opened, the access time to it is identical to that of system volume files.

- Many software products do not allow for private volumes in their installation procedures.

  You *may* need to modify the installation procedures of these products. However, once the software is installed, it will run perfectly well on private volumes.

- STORE and RESTORE only deal with groups and not volume sets as a whole.

  To STORE all of the files from a particular volume set, all of the groups and/or accounts that reside on that set must be enumerated. (You may want to do this using an indirect file.)

TERMS AND DEFINITIONS

This next section provides a definition of terms and concepts necessary to an understanding of how private volumes work. Following this, we'll examine how they are implemented.

VOLUME SET

A set of disc volumes treated as a unit by MPE. A private volume set consists of exactly one master volume (which contains the set's file directory) and zero or more 'slave' volumes, used to expand the storage capacity of the set. In general, for any volume in the set to be accessed, all other members of the set must be online.

SYSTEM VOLUME SET

The volume set defined by the system manager in :SYSDUMP's VOLUME TABLE CHANGES dialogue. This must include ldev 1, which is this set's master volume. MPE requires that the entire system volume set be online at all times.

PRIVATE VOLUME SET

A volume set other than the system volume set. The set contains exactly one master volume and zero to seven slave volumes. It is defined using the :NEW/:ALTVSET commands by a user with CV (Create Volumes) capability.

MASTER VOLUME

The master volume is the volume of a volume set which contains the set's file directory. It has the same name as the volume set. Each volume set contains exactly one master volume.

SLAVE VOLUME

A slave volume is a volume of a volume set that does not contain a file directory. It is simply a volume added to the set to increase the storage capacity of the set.

HOME'VOLUME'SET

The volume set on which a group's files are stored. A group must reside wholly on a single volume set (private or system), but different groups in an account may reside on different sets. The home volume set is specified via the ;VS= keyword of :NEW/:ALTGROUP commands.

VOLUME CLASS

A volume class is a subset of a private volume set, always including the set's master volume. Groups may be assigned to volume classes, and if so only the members belonging to the class must be available to access the group's files. A volume class is the smallest unit of a volume set which must be online in order to access some of the set's files.

PHYSICAL MOUNT

Placing a disc pack in a drive and setting the drive online. This is indicated by a message on the system console.

LOGICAL MOUNT

Reserving a physically-mounted volume for use. This can be either explicit (:MOUNT command) or implicit (FOPEN, :STORE). BINDING (see below) occurs for an implicit mount and is optional on an explicit mount.

SPANNING

The creation of account or group directory entries on the master volume of a private volume set. This step is required before files in the group can be created on the volume set. Note: this term is often confused with binding (see below). Spanning is invoked by the :SPAN option of the ;VS= parameter of the :NEW/:ALT ACCT/GROUP commands.

BINDING

Binding is the operation performed by the system to link temporarily the group entry in the system directory to a group entry on a master volume. This is generally invisible to users.

IMPLEMENTING AND USING PRIVATE VOLUMES

Now that you have a basic understanding of PVs and the terminology surrounding them we can proceed with a discussion of how to implement and use them on your system.

Configuring Private Volumes

To configure a PV the System Manager proceeds precisely as if configuring an ordinary system domain disc. The only difference is that he *does not* make an entry for the disc in the SYSTEM VOLUME TABLE. (Entries are put into the SYSTEM VOLUME TABLE in the "VOLUME TABLE CHANGES" function in the :SYSDUMP/:INITIAL dialog.) The way that MPE determines if a disc is in the SYSTEM DOMAIN is by seeing if its VOLUME ID (more about this shortly) is in the SYSTEM VOLUME TABLE.

Creating Volume Set Definitions

We just said that a PV is configured the same as a regular system domain disc except that no entry is made to the SYSTEM VOLUME TABLE. Without an entry in this table MPE will allow the device to be used as a foreign, serial or PV disc. So how do we tell the system we want it to be used as a PV? We create a VOLUME SET DEFINITION with the :NEWVSET command (set definitions are altered with the :ALTVSET command). The :NEWVSET command requires CV (Create Volumes) capability. Use of this command is, to Private Volumes, what making entries into the Volume Table is to the system volume set. Using :NEWVSET, we define:

- The name of the volume set
- The name of each member of the set (up to 8)
- The type of disc drive used for each member

One of the member volumes must be defined as the "master" volume. This member must have the same name as the whole volume set and will contain the account/group/file directory for the entire set.

This next point may be confusing, so be careful: the information entered using :NEWVSET is recorded as a Volume Set Definition (VSD) in the system directory (LDEV 1) under the logon group and account. It's important to remember that what we're doing here is creating the DEFINITION of the set -- no more, no less. What we are NOT doing is in any way affecting the placement of files which belong to the logon group. The location of the VSD in a particular group/account in no way affects what groups/accounts can be placed on that volume set, nor which users/accounts can access such files.

For example, if the following commands are entered:

```
:HELLO MANAGER.SYS,PUB
:NEWVSET ADMIN;MEMBERS=ADMIN:HP7935,ADMIN1:HP7935
```

the effect will be to create a two-volume set called ADMIN.PUB.SYS, with a master volume ADMIN and one slave volume ADMIN1.

Private Volume sets may be expanded to include more slave volumes after they've been defined. This is done via the ;EXPANDSET= parameter on the :ALTVSET command. However, member volumes may not be deleted. To delete volumes, the entire definition must be purged and recreated. This will render any existing volume set of that name unusable if the new set is smaller than the old.

A Volume Set Definition cannot be deleted while that set is in use. If the set is not in use and the definition is accidentally deleted, simply recreate it exactly as it was and the set itself will be usable again. MPE will permit a VSD to be expanded even though the set is in use, but this will create havoc and should not be done.

Volume Set Definitions can be deleted by their creators with :PURGEVSET, and anyone can list them using :LISTVS. :LISTVS is similar to :LISTF in that it permits 'wild card' characters and provides three levels (0, 1, and 2) of detail.

Defining Volume Classes

Volume Classes are subsets of private volume sets and consist of the set's master volume plus zero or more of its slave volumes. These can be used to restrict a group to certain volumes, and to reduce the number of volumes which must be on-line to access a particular group's files.

Volume Classes can be defined at the same time as the volume set, using the ;CLASS= parameter of :NEWVSET, or they can be added on later using the ;ADDCLASS= parameter of :ALTVSET. They can also be expanded (like sets can) using ;EXPANDCLASS= of :ALTVSET.

As with their parent volume sets, volume classes are deleted with :PURGEVSET and can be listed with :LISTVS. Deleting a volume SET definition DOES NOT delete its associated volume CLASS definitions; these must be deleted explicitly. :LISTVS identifies a volume class with a "(C)" next to the class name.

Volume Classes are also stored in the directory, in the same format as Volume Sets. There is a flag in the entry indicating whether it is a class or set definition.

After a private volume set is defined, its volumes must be initialized using :VINIT (PVINIT.PUB.SYS). This is done by first formatting the pack and then using the INIT command to place a private volume label in sector 0. This command also creates a Disc Free Space Map and, if the pack is the master volume, the directory.

To initialize the volumes of ADMIN.PUB.SYS described earlier, enter the following:

```
:DOWN 3
:DOWN 4
:VINIT
>FORMAT 3
>INIT ADMIN,3,ADMIN.PUB.SYS
ENTER DIRECTORY SIZE (SECTORS): 512
>FORMAT 4
>INIT ADMIN1,4,ADMIN.PUB.SYS
```

```
  >EXIT
  :UP 3
  :UP 4
```

It's important to remember that this procedure will cause loss of all data currently on the specified disc. :VINIT will ask for confirmation if you try to format or initialize an existing private volume.

Creating Entries in Private Volume Directories

Before files can be placed on a private volume set, three things must happen:

- The account and group must exist in the system directory.
- The group must have a Home Volume Set (HVS) specified in the system directory.
- The account and group must exist in the PVs directory.

The key here is understanding the operation of the ;VS= keyword of the :NEW/:ALTxxxx commands. This keyword performs two functions:

- Creation of Home Volume Set (HVS) assignment, for groups.
- Creation of entries in the PV directory, for both groups and accounts. This operation is called SPANNING.

Let's assume we have an account called PVCLASS and one group in it called PUB. In order to have the files of PUB.PVCLASS reside on the volume set ADMIN.PUB.SYS, we first need to create an account entry for PVCLASS on the volume set. This is done as follows:

```
  :ALTACCT PVCLASS;VS=ADMIN.PUB.SYS:SPAN
```

The SPAN keyword tells the system to create an entry for this account on ADMIN.PUB.SYS. Since accounts do not have home volume sets, the only function of the ;VS= keyword for :NEW/:ALTACCT commands is to create (or alter) the account entry on the specified volume set; you must specify :SPAN (to create an entry) or :ALT (to change an existing entry).

Now that the account is created, we must perform the HVS assignment for PUB and create an entry for it on the volume set. This is done as follows:

```
  :ALTGROUP PUB;VS=ADMIN.PUB.SYS:SPAN
```

In this case, the ;VS= keyword is performing two different functions: first, it is recording the name of the group's home volume set in the group's system directory entry; and second, because the :SPAN option was specified, it is creating a group entry in the private volume's directory. If one were to leave :SPAN off, the only thing that would happen is that the name of the Home Volume Set for this group would be changed in the group's system directory entry (presumably the group entry would already exist on the corresponding private volume, as when transporting the volume to a different system).

Note that as a result of these operations, each private volume group will have two directory entries: one in the system directory, and one in the private volume's directory. Each account which has groups on private volumes will have one system directory entry, plus one entry on each private volume on which it has any groups. Where then, does MPE get information such as file space limits or capabilities, with so many sources from which to choose? The following table specifies from which directory MPE obtains its information:

From System Directory

- Capabilities
- Local attribute
- CPU limit & count
- Connect limit & count
- Max priority
- UDC information
- Passwords
- HVS name (groups only)
- R,W,A,L,X user types

From PV Directory

- File space limit
- File space count
- SAVE access user types

The normal directory listing commands (:LISTxxxx, :REPORT) behave differently with private volumes. :LISTxxxx only shows the system directory entry; to examine the private volume entry, you must use LISTDIR5.PUB.SYS. :REPORT provides a ;VS= option so that you can tell it which directory you want. However, since the information this command displays is actually split across the two directories, in order to get it all you must issue two :REPORT commands. One should be without the ;VS= keyword (which will show CPU and CONNECT information, but zero for file space information) and one should be with ;VS= (which will show file space information, but zero for CPU and CONNECT).

Similarly, :PURGEACCT and :PURGEGROUP provide a ;VS= keyword. Without it, only the system directory entry is removed; the private volume directory (and files) remains unchanged. When ;VS= is used on these commands, the private volume directory entry and files are removed, but the system directory entry remains. Therefore, you must issue two :PURGExxxx commands to remove an account or group completely.

QUICK REFERENCE

- The private volume cannot reside in the system domain. (e.g. It's volume name doesn't appear in the disc based copy of the system volume table.) However, if you were to look at the memory copy of this table you would see the PV volume name when it is recognized by MPE.

- You *never* FORMAT a CS80 disc. These include 7911, 7912, 7914, 7933, 7935, 7936, 7937, 7945, 7957, and 7958.

- To convert system domain discs to private volumes takes a RELOAD, because you must delete volumes from the volume table.

- Converting a private volume to the system domain can be done with COOL, COLD, RELOAD, or UPDATE options.

- Moving a group from a system domain disc to a private volume (and vice versa) requires a :STORE followed by an :ALT(ACCT/GROUP) and then a :RESTORE.

- The volume *must* be conditioned before the private volume definition can be placed on it. See the VINIT, INIT, FORMAT, and SCRATCH commands.

- The operator *must* issue a :VMOUNT command to make the private volume facilities available to the system users.

- If the ",AUTO" is omitted from the :VMOUNT command the operator will be prompted for mount messages.

- You *must* use the ;VS= on the :REPORT command to find the disc space in use for accounts/groups which are spanned (:SPAN) to the private volume set.

- You *must* omit the ;VS= on the :REPORT command to find the cpu and connect time associated with a spanned private volume account/group.

- You *must* use the :SPAN parameter at the account level if you want one or more groups in the account to reside on a private volume set.

- You *must* use the :SPAN parameter on the group if you want the files in that group to reside on the private volume.

- A group *can not* have files in both the system and private domains. If the group is spanned *all* files will be in the private volume. If the group is not spanned *all* files will be in the system domain.

- An account which is spanned can still have groups in the system domain. Likewise an account which has *not* been spanned cannot have groups in the private volume.

- When moving private volume disc packs from one system to another the directory structure on the receiving system *must* have the necessary accounts & volume definitions.

- Purging accounts or groups requires two purge commands to completely delete the directory structures from both the PV and the system domain.

  ```
  :PURGEACCT PAY;VS=EXAMPLE.PUB.SYS
  :PURGEACCT PAY
  ```

- Virtual memory can *not* be allocated on a private volume disc.

- You *must* have CV, create volume, capability to create new volume set definitions.

- Users *must* have UV, use volume, capability to access private volumes.

COMMANDS ASSOCIATED WITH PRIVATE VOLUMES

ALTACCT

The ALTACCT command is used to alter the attributes of an existing account entry. For private volumes the ;VS= and :SPAN parameters are used to have the system perform the necessary directory maintenance on the private volume.

ALTGROUP

The ALTGROUP command is used to alter the attributes of an existing group within an existing account. For private volumes the ;VS= parameter is used to designate a private volume set for the group. However it is necessary to use the :SPAN in order to force data files associated with this group to the PV. Note there must be no files in a group to be SPANned. You must first STORE and PURGE the files.

ALTUSER

The ALTUSER command is used to add UV capability to users who will be accessing private volumes. UV capability is required to access any files on private volume sets.

ALTVSET

The ALTVSET command is used to add class names and members to an already existing private volume definition.

DISMOUNT

This command allows you to dismount a volume set/class which was previously explicitly mounted via a :MOUNT command.

DOWN

Allows you to remove a device from service. For private volumes this command is used to insure no users are accessing the volume prior to swapping the disc pack. Additionally it is used prior to many VINIT functions. It only needs to be issued once. The inverse command is :UP.

DSTAT

This command will display the status of the disc drives on the system. For private volumes it will provide the volume set name and generation index also.

LDISMOUNT

Allows the operator to logically dismount a private volume set/class The set/class will remain unavailable until the next LMOUNT/MOUNT command.

LISTVS

This command will produce a formatted listing of a volume set/class definition for you. A LISTVS @.@.@,2 will list all volume sets and class definitions on the system.

LMOUNT

The LMOUNT command allows the operator to logically 'lock' a previously mounted private volume in place. That is to say the volume set will remain mounted between accesses and between job/ sessions.

MOUNT

This command requests that the system operator mount a volume set/ class. Unless there was a previous :VMOUNT ON,AUTO issued this command will generate an operator request which must be replied to.

NEWACCT

This command will place an entry into the system directory creating a new account. When used with private volumes and the ;VS= parameter is used and a :SPAN is present, the necessary directory maintenance is performed on the private volume set also.

NEWGROUP

The NEWGROUP command will create a group in an existing account. When used with private volumes and the ;VS= parameter and the :SPAN is present it will cause the group directory entry creation in the private volume directory also. All files created in this group will now reside on the private volume.

NEWVSET

This command allows the user with CV capability to define a new private volume set. It allows the user to define the volume set, members, and classes.

PURGEACCT

When the PURGEACCT is issued and the ;VS= parameter is present the system will remove the data files, groups, users and the account information from the directory on the private volume. But the entry still exists in the system domain directory. To clear the system directory you need to issue another PURGE command *without* the ;VS= parameter.

PURGEGROUP

When the PURGEGROUP command is entered and the ;VS= parameter is present the data files and group information is removed from the directory on the private volume. To remove the group entry from the system directory will require another PURGE command WITHOUT the ;VS= parameter.

PURGEVSET

Deletes an existing volume set/class definition from the system. You *must* enter a PURGEVSET command for all class definitions and the volume set definition to completely eliminate the entire PV definition from the system.

REPORT

When the report command is entered and the ;VS= parameter is present it will display the relevant MPE accounting information stored in the directory on the master private volume.

NOTE with the REPORT command disc space used is tracked in the directory on the private volume and will only be displayed when the ;VS= parameter is used. However, the CPU and CONNECT times are maintained in the system directory and the ;VS= should NOT be used to get these values.

UP

The UP command is used to make a device available to the system. It is the inverse of the :DOWN command. The system will perform an AVR, automatic volume recognition, on the device and the label/ volume set information can then be viewed with the DSTAT command.

VINIT

The VINIT command will place you into the private volume management utility.

VMOUNT

The VMOUNT command is used to enable/disable the MPE private volume facility. If the ON parameter is used and the ,AUTO is not present then the operator will be required to :REPLY to each request for access to any private volume. If the ;ALL parameter is present then all console related messages associated with private volume accesses will be printed. An example of the command to make all messages print and have MPE automatically satisfy all requests that it can would be: :VMOUNT ON,AUTO;ALL

VSUSER This command will format and print a list of all users currently accessing a mounted volume set.

QUESTIONS & ANSWERS

Q. I performed a PURGEGROUP command and forgot to put the ;VS= parameter on the command. What do I do to get the group purged from the private volume set?

A. Redo the purge command and supply the ;VS= parameter this time.

Q. I performed a PURGEACCT command and forgot to put the ;VS= parameter on the command. What do I do now to get the account purged from the private volume set?

A. Redo the purge command and supply the ;VS= parameter this time.

Q. I executed a PURGEVSET command before I realized it was necessary to purge the accounting structure first. What happened and how can I get the system back to a known state without a RELOAD?

A. Just rebuild the volume set definition. The disc packs are not touched by the PURGEVSET command.

Q. How do I get a disc out of the system domain so I can make it a private volume?

A. Perform a full backup of the system. Delete the volume (only) from the volume table using SYSDUMP. Spin the disc down and perform a RELOAD. After the system comes up spin the disc back up and prepare the media as stated in the System Operation and Resource Management Manual.

Q. What is the recovery from a head crash on the master volume?

A. You must recreate the entire volume set after the disc has been repaired.

Q. What is the recovery from a head crash on a slave volume?

A. You must recreate the entire volume set after the disc has been repaired.

Q. How do I designate the master volume?

A. The master volume will be the volume whose member name is the same as the name of the volume set in the NEWVSET command. For example:

```
NEWVSET myset;MEMBERS=myset:HP7935,slave1:HP7925
```

In the above example 'myset' will be the master volume.