HSM - Hierarchical Storage Manager

Overview:

HSM is an add-in to the VMS I/O system which provides VMS with the ability to have transparent file migration between active and near line storage in one or more steps. When files are migrated ("shelved") from normal disk storage to backing storage, a marking is left on them which is automatically read so that when the file is opened, it is automatically retrieved by HSM from nearline storage. Thus, a user or program need not be aware at all that such shelving occurred and no operator intervention is needed to perform the "unshelving" operation. This is distinguished from operation where a user must first request an archived file be reloaded, which requires detailed advance knowledge of such needs. The files appear to have been on disk all the time, but in fact the online disk space is conserved. It is also distinct from modes of operation where a file's location visibly changes. These, too, require that programs be told where the new site is, which can be awkward. HSM provides total transparency of file migration, invisible to programs and users apart from small delays where files must be unshelved.

In addition, HSM provides two unique "soft link" abilities which complement unshelving, and manages volume space

The basic capabilities of HSM are these:

* Files can be shelved (by space-making script or by command) and unshelved automatically from nearline storage when they are opened.
The process opening the files then sees a successful open with no side effects. Shelved files can be stored in compressed form if this is desired, and can be stored in any desired location. Storage of shelved files on tapes or the like can be done also. (In Version 1, users must modify command files to do this.)

* Files can be "soft linked" to other files, even across disks. This mode of access can be used for a sort of permanent shelving on another volume by truncating the original file to zero blocks. The soft link operates extremely fast and causes the file in question to be opened in its new location, with the

channel restored on close so that again a program observes no change, but the file is accessed transparently at its new site. Where the new site is a read/write device this can be most effective.

* Files can be "soft linked" in a "readonly" mode to another file. In this mode, suitable for read-only backing storage, whenever a file is opened for read-only access, it is transparently and instantly opened on its linked site, wherever that may be on nearline storage. When such a file is opened for any kind of writing, however, it is treated as a shelved file and is unshelved and replaced on normal disk before the open is done. Thus any read/write access will find the file in a suitable location for its open to succeed, transparently. (Notice that soft linked storage must be on disks and must not be in compressed form. The decompression would induce an unwanted delay in access.)

* Disk space can be managed. Whenever an extend or create (or inswap) would not have adequate space on disk, HSM starts a "make-space" script which is tailored by a fullscreen utility to match site policy. This policy can select files based on access time, size, name, or characteristics for space making. The default policy is to shelve files over a week old.

Installation

The HSM package is installed in a three step process. First, ou install HSM with VMSINSTAL to build the images and put them somewhere. Then you use HS_SETUP.COM to generate the startup scripts and select features. Finally, you may optionally elect to use HSSPACE_TAILOR.COM to set site policy for files to be selected for shelving when space must be reclaimed. This step is not however required.

1. VMSINSTAL
First, HSM is installed with VMSINSTAL. Put the medium with the HSM kit on it and give the command

```
@SYS$UPDATE:VMSINSTAL HSM010 medium:
```

Only one non-obvious question is asked:

```
Enter directory for HSM programs
```

The response to this is where the HSM programs and command files will be installed. It should be accessible to the whole cluster if you have a cluster, and is where the later setup script should be told they are.

2. HSM_SETUP

Next, you run the command

```
$ @SYS$MANAGER:HS_SETUP
```

to perform your system setup. This allows you to set the system parameters up and will generate the command files to start HSM up at boot time and to create a few useful symbols at login.

The setup script initially comes up with a menu which looks like this:

```
    HSM CONTROL/MAINTENANCE 15:44:05

    *Select "Shelving" style HSM
     Select "softlink" style HS
     Select "softlink" style HSM
     Select "readonly softlink" style HSM
     Explain the HSM styles briefly
 --> Proceed to set up HSM configuration file
     Unshelve all files on a disk if space exists
     Stop a running HSM server cleanly
     Start a server that is already configured
     Quit, do nothing
```

This menu is intended to provide a top level setup control which might be used for installing or deinstalling HSM. Normally in an HSM installation you simply take the first selection, since the other items are for use once HSM is installed and configured.
The rest of the menu has the following meanings:

`Select "Shelving" style HSM"`, `"Select "softlink" style HSM"` and `"Select "readonly softlink" style HSM"` specify which operation is to be performed to make space on the disk when the space-finding script is run. In all these cases, files are placed on some nearline storage (which means, some local disk including compressed disks managed by Squash or magneto optical disks) and their space reclaimed on the main disk. This reclamation is done by truncating the file to zero blocks and marking that HSM will perform an action when someone tries to open the file. In the case of "shelving" style operation, the operation performed is to delay the open until the file is copied back to the normal disk, then let it by. In the case of "softlink" operation, the file is copied to some other nearline disk and is opened there, invisibly to the program, instead of on the original disk. This open takes place very quickly and the file is never brought back to the original disk, just used in place wherever it has been put. To reset the user program's I/O channel to point to the original device, it is necessary that HSM do a minimal management of the nearline disk also. This is done (as explained below) by selecting the nearline disks but telling the configuration script that they are only nearline destination disks, not disks from which softlinks or swapping can take place. If the disks are managed for full function HSM, this restricted marking is not necessary. In the case of "readonly softlink" style HSM, the operation depends on the type of open. If the file is opened for read only, the file opens on the nearline storage. If the file is opened for read/write access, however, it is unshelved to the original disk from whence it came and the marking is removed so subsequent accesses find it on the original disk. At this point it is also deleted from the nearline disk.

"`Unshelve all files`..." is used where the desire is to deinstall HSM. In this case one must unshelve any files which have been shelved (placed in secondary storage) by HSM since once HSM is deinstalled, it can no longer automatically retrieve them. This script can only succeed if there is room enough on the disk to allow such unshelving and will not begin unless this is the case. It will however unshelve all files on a given disk so that HSM can be stopped and removed, provided this is possible to do cleanly.

"`Stop a running HSM server`" will ask which disk to remove the HSM server for and will cause that server to exit cleanly. Note that if a server controls several disks, HSM is stopped for all of them, not just one.

"`Start a server that is already configured`" simply starts any HSM servers that are started by the HSM startup script. It can be used after stopping a server if it is desired to restart it.

"`Quit, do nothing`" simply bails out of the entire HS_SETUP script.

Once the initial menu has been run, and you select the item to "Proceed...", you will be presented with the following menu:

```
    HSM SETUP  19:19:35

   -->  Set area to hold HSM database files
        Set start intercept driver unit number (now 0)
       *Set area for HSM executable images
        Done this menu, process disk selection
        Remove a disk from an existing HSM configuration
        Set images which are exempt from HSM (e.g. defraggers)
        Set area for storing nearline files shelved
        Quit, do nothing
```

Notice that the "Set area for HSM executable images" is tagged with an asterisk (*). This indicates that a logical was found giving this value. This will usually be the case after a VMSINSTAL run. If definitions exist for other storage areas, they are also tagged with "*" to indicate it is not necessary to enter them again.

The area to hold HSM database files, the area for HSM executable files, and the area for storing nearline files shelved should all be filled in before selecting the "Done this menu..." item.

The menu items have the following meanings:

`Set area to hold HSM database files`
HSM stores data in a database of its own indicating where files were stored. Each disk has one (and for volume sets one uses `set file/enter` to make several synonyms for a file). Whenever a file is to be shelved or unshelved, this file is consulted, and locks on it serve also to synchronize clus-

ter file access at these times. The area specified will hold these files. It can be anywhere in the system, though it should be a local disk.

`Set start intercept driver unit number (now 0)`

You can have as many HSM servers as you like, where each server controls unshelving and space monitoring for one or more disks. Each disk uses one unit of the intercept driver (HSDRIVER). To set up a second (or later) server, you may start using HS units higher than 0, so that if you set up a server once for three disks (thus using HSA0:, HSA1:, and HSA2:), you can set the start unit number to 3 here to start with HSA3: next time. Each run of HS_SETUP.COM generates control files for one server (appending to the startup files if it is not the first time). Normally, it is expected that inswap activity will be low enough that one HSM server will do for an entire system. Thus, this is the only provision for multiple servers. You must track how many disks have been set up for by hand if you choose to have multiple servers.

`Set area for HSM executable images`

This area is expected to contain the images for the HSM server, control utilities, and command files used by the system. It is asked for by KITINSTAL and normally would just be left as it was set.

`Done this menu, process disk selection`

This item allows you to go on to picking disks.

`Remove a disk from an existing HSM configuration`

This item allows you to remove disks in an existing HSM configuration (or incidentally find out what disks are configured currently).

`Set images which are exempt from HSM (e.g. defraggers)`

Some images should not be able to generate inswaps. Defragmenters and the like can be in this class. Also it is often a good idea to have a copy of the BACKUP image, pointed to by its own verb, which will not cause unshelving. Shelved files are marked as NOBACKUP type to prevent a system backup from unshelving all files on a disk, but this can be changed if one does system backup from an image in the "exempt" list. Up to 32 image pathnames can be specified. (The pathname to use can contain wild cards, e.g. `"*]BACKUP.EXE;*"`, or use the full path returned by $GETJPI (as seen, for example, in a `SHOW PROCESS/CONTINUOUS` display) as the equally.) When an image is exempted, the HSM server simply allows the image's file opens to proceed with no change.

`Set area for storing nearline files shelved`

This menu item sets the area on nearline storage where this server will store files. It should be of form device:[directory] since the file ID is all that is really available to the HSM server. This sets the logical name used for this area by the .COM files. Note that you will be warned if the area cannot be written to. Also note that if it is desired to define this area differently, the command files can of course do so, or an Assign/job where the servers run would do this. The current setup assumes a single location for the system.

When the initial setup menu is exited, one goes on to select disks to be enabled. Remember that for soft links to work properly, the disk on which the link is AND the disk on which the file linked TO is must both be controlled by HSM.

Disks selected may not, in the initial version of HSM, be system disks. The menu will display all disk class devices found, as well, but it should be noted that only local disks should be selected for HSM. Use with nonlocal disks might or might not work, but is not supported. ("Nonlocal disks" means things like NFS clients and the like, not disks on an HSC, which are local for purposes of HSM.)

The disk menu looks like this (with the selected disk highlighted by reverse video:

```
HSM Configuration
          (Non system)  Disk Selection
   Use arrows to move to selection. Use RETURN to select.

End disk selection
_ARISIA$DKA700:
_ARISIA$DKB0:
_ARISIA$DKB300:
_ARISIA$DCA0:
_ARISIA$DCA2:
_ARISIA$DCA3:
_ARISIA$DCA4:
_ARISIA$DCA5:
_ARISIA$VDB0:
_ARISIA$VDB1:
_ARISIA$DKB200:
_ARISIA$DKB700:
_ARISIA$DCA1:
_ARISIA$DCA6:
_ARISIA$DCA7:
_ARISIA$FQA0:
_ARISIA$FQA1:
_ARISIA$FQA2:
                                              Currently on item 1 of 151
```

In this list, which will scroll as you move the arrow keys around, mounted disks on the system are shown first, then all other disk class devices. When a disk has been selected, it is tagged with * on the line to flag visually that you've already set up HSM for that disk.

If you have selected an HSM style other than "shelving", you are also asked after selecting a disk whether this disk is just a nearline disk which will be used as a place to store softlinked files. If this disk is the target of softlinks, but will not have any of its files softlinked anywhere, nor shelved, you should reply "Y" to the following question. (Note: Select the input disk or disks first when selecting disks.)

```
Is this disk only a nearline disk destination [Y/N][N]:
```

If you answer Y to this question, this disk will be set up so that HSM can replace I/O channels for softlinks, but nothing else.

After possibly answering the above question, you return to select another disk and eventually to exit. If the "Quit..." option is selected, you quit out of the whole disk select menu, not just this disk.

3. HSSPACE_TAILOR

The HSM function of monitoring space depends on a site policy to decide what space should be freed. This site policy is decided upon in the HSSPACE_TAILOR.COM script. The MAKSPC.COM procedure is furnished with HSM so that additional site constraints can be added should they be desired. The space tailoring function is used if you have enabled the function to try to make space when a disk runs out of room. The space is found via a script which must embody your site policy for what files to shelve and what may not be. It will respect a /NOSHELVE attribute or a /NOMOVE attribute on VMS V6 and later, but you must set its limits and in addition if you wish to exempt certain files from moving, you must set up the commands which permit such selection.

When you run this script with

```
$@GCY$SYS:HSSPACE_TAILOR
```

you will create a space-making script for your system.

The script produces a menu which looks like this:

```
Site Space Policy Determination

--> Set min size file to migrate when space is needed (now 100)
    Set min age in days for files to be moved (now 7)
    Set largest "medium sized" file size (now 1000)
    Specify substrings of file paths to leave alone
    Clear substring list (now "")
    Report space occupied by files which would be selected
    Run the space-making script to free max space
    Done this menu, build the selection script
    Quit the menu, doing nothing
```

The script selects files based on age and size, so that files whose last access is at least **N** days ago and whose size is at least **M** blocks and which don't contain strings in the full file specification that you specify may be shelved. Files marked installed, contiguous files, anything with `DSK` or `SYS` in the filename string, ISAM files, or `NOBACKUP` files are not considered eligible.

You may impose other filename strings that cannot be there. For example, to omit any file pathname containing the string `SQUASH`, add "`SQUASH`" to the substring path list. The minimum age and size are set with the first 2 menu items. You can of course quit the menu at any time.

The space selection will normally select files which have not been accessed (as determined by the latest of creation, revision, or expiration dates) for 7 days. The files selected are, first, those of sizes 100 to 1000 blocks; next, those of 1000 to 250000 blocks; last, those of 10 to 100 blocks. File movement continues until sufficient space is found for the current operation (plus a small margin) and then stops. Note that this controls the automatic movement of files only. If you use the file shelving/softlinking menu (see below) you can enter softlinks or shelve any files you select, based on your selection.

The "Report..." item will generate a report showing how many files in each of the size classes (medium, large, and small) are available for making space, and the "Run the space-making script..." item will run the space making script, attempting to free up to 10% of the disk space. Since some scans of the entire index file are needed to gather this data, be prepared for the information to take sometimes a few minutes to gather.

Whenever a disk under HSM control is nearly full and a file create or extend for less than 1/8 of the volume size is seen, then the file `GCY$SYS:MAKSPC.COM` is run. It is operated as follows

```
$ @GCY$SYS:MAKSPC devicename: nblocks
```

and will attempt to free nblocks blocks on devicename: before returning. It does this by outswapping files that match the site policy you have just chosen. Note that the site policy will use the later of the expiration date, the creation date, or the revision date of a file. Thus if you have used the

```
$ SET VOLUME /RETENTION=(1-,2-)
```

command so the system records the last day of file access, the selection will skip files which have been read recently, even though they might not have been written for some time. This is recommended.

It is also suggested to run the make-space script at night so that the outswapping or shelving can be done out of prime time, rather than impose a delay on file opens as space is found. This should be done via a repeating BATCH process which will resubmit itself for the next day when done. If however you have any scheduling packages, such a package can be used instead of a batch submit. This is not required for operation, but is likely to be a useful thing to do.

Now the basic installation should be complete. You may of course rerun the `HS_SETUP` at any time to alter decisions encoded there.

SHELVING FILES BY COMMAND

There are two commands to do manual shelving. One of these lets you use a fullscreen front end to select files in your current directory tree. It is the `MOVEHSM` command. The other lets you specify a file specification (possibly a wild card) on any device for shelving. It is the `MOVEFILE` command.

There are several modes of file shelving or moving, and these commands allow them all to be selected.

The `MOVEHSM` command presents a fullscreen directory of files, with a fairly large set of possible formats. You use arrow keys to move around and select the file by the space bar (or deselect the same way) or use other selection criteria to pick pieces of filenames.

Such a screen has the following appearance:

```
Edit   eXecute   Copy   Rename   Delete    Move   pUrge   Quit   Help   List   ?
EVERHART                    USR$ROOT:[EVERHART.HSM]                  30 Sep 22:13
+----------------------------q FILE MANAGER  *.* ------------------------------+
!  !HS.MMS;2                   7/8   7-SEP-1994 19:45  (RWED,RWED,RE,RE)     !
!  !HSACLCREA.TXT;2            1/2   7-SEP-1994 19:45  (RWED,RWED,RE,RE)     !
!  !HSALL.BLD;3                2/2   7-SEP-1994 18:05  (RWED,RWED,RE,RE)     !
!  !HSALLDBG.BLD;2             2/2   7-SEP-1994 19:45  (RWED,RWED,RE,RE)     !
!  !HSAUTH.FOR;2               3/4   7-SEP-1994 19:45  (RWED,RWED,RE,RE)     !
!  !HSAUTHM.MAR;2            10/10   7-SEP-1994 19:45  (RWED,RWED,RE,RE)     !
!  !HSAUTHM.OBJ;1              3/4   9-SEP-1994 19:05  (RWED,RWED,RE,RE)     !
!  !HSAUTHMAINT.DOC;8        17/18  14-SEP-1994 17:56  (RWED,RWED,RE,RE)     !
!  !HSAUTHMAINT.EXE;1        35/36   9-SEP-1994 18:09  (RWED,RWED,RE,RE)     !
!  !HSAUTHMAINT.FOR;9        56/56   8-SEP-1994 19:39  (RWED,RWED,RE,RE)     !
!  !HSAUTHMAINT.OBJ;1        35/36   9-SEP-1994 17:08  (RWED,RWED,RE,RE)     !
!->!HSDMN.CLD;2                3/4   7-SEP-1994 17:45  (RWED,RWED,RE,RE)     !
!  !HSDMN.EXE;1              61/62   9-SEP-1994 18:09  (RWED,RWED,RE,RE)     !
!  !HSDMN.MAR;3            100/100   7-SEP-1994 19:48  (RWED,RWED,RE,RE)     !
!  !HSDMN.OBJ;1              19/20   9-SEP-1994 18:06  (RWED,RWED,RE,RE)     !
!  !HSDRIVER.DOC;7             4/4   7-SEP-1994 18:08  (RWED,RWED,RE,RE)     !
!  !HSDRIVER.EXE;1           21/22   9-SEP-1994 18:09  (RWED,RWED,RE,RE)     !
!  !HSDRIVER.MAR;10        276/276  30-SEP-1994 17:04  (RWED,RWED,RE,RE)     !
!  !HSDRIVER.OBJ;1           24/24   9-SEP-1994 19:05  (RWED,RWED,RE,RE)     !
!  !HSDRIVER.OPT;2             1/2   7-SEP-1994 18:46  (RWED,RWED,RE,RE)     !
!  !HSDRIVER_S2.MAR;19     288/288  30-SEP-1994 17:05  (RWED,RWED,RE,RE)     !
!  !HSENTER_PASSWORDS.COM;2    3/4   7-SEP-1994 18:46  (RWED,RWED,RE,RE)     !
!  !HSEXEDEL.CLD;2             1/2   7-SEP-1994 17:46  (RWED,RWED,RE,RE)     !
!  !HSEXEMPT.CLD;3             2/2   7-SEP-1994 18:02  (RWED,RWED,RE,RE)     !
!  !HSEXEMPT.EXE;1           30/30   9-SEP-1994 18:09  (RWED,RWED,RE,RE)     !
!  !HSEXEMPT.MAR;2           40/40   7-SEP-1994 17:46  (RWED,RWED,RE,RE)     !
!  !HSEXEMPT.OBJ;1             5/6   9-SEP-1994 18:06  (RWED,RWED,RE,RE)     !
!  !HSFILDEL.COM;2             3/4   7-SEP-1994 19:46  (RWED,RWED,RE,RE)     !
!  !HSFILEMARK.COM;2         14/14   7-SEP-1994 20:46  (RWED,RWED,RE,RE)     !
!  !HSFILRST.COM;2             5/6   7-SEP-1994 19:46  (RWED,RWED,RE,RE)     !
!  !HSFILSAV.COM;3             4/4  29-SEP-1994 18:27  (RWED,RWED,RE,RE)     !
!  !HSFILUNDEL.COM;2           2/2   7-SEP-1994 17:46  (RWED,RWED,RE,RE)     !
!  !HSGETLPORT.MAR;2           3/4   7-SEP-1994 18:46  (RWED,RWED,RE,RE)     !
+------------------------------------------------------------------------------+
! current file no 12    total files 71    files selected 4    display format 2 !
+------------------------------------------------------------------------------+
```

When you type the character G (or g) you are given the following prompt:

operate on the 4 selected files Y/N [N]

If you reply Y, then you enter the file shelving menu.

If you enter via the `MOVEFILE` command you will be asked what file(s) to move. Then in either case you enter the shelving menu.

The file shelving menu looks like this:

```
      Hierarchical Storage Facility - File Moving
            File USR$ROOT:[EVERHART.HSM]HSDMN.CLD;2

    -->  Set normal softlink to another file
         Set R/O softlink to another file, moving this there
         Mark and move this file now to backing area
         Quit, make no changes to this file
         Perform database maintenance commands
         Move and mark all selected files now
```

This selects what you can do to the file. The three modalities of use are represented. In addition you can do maintenance commands including resetting the file of kernel markings.

NOTE WELL:

In this menu, the marking commands work at once, so ***once you select any of the first three items, it is too late to quit and leave the file alone.***

The three types of marking in the menu work as follows:

Recall, there are three types of linking that can be used.

1. You can create a normal "softlink". A softlink means that when your file has one, it also has a pointer to another file on possibly some other device. Whenever someone attempts to open your file, the other file is opened instead on the other device, invisibly, and with no need for the HSM server even to be involved. As long as the device the file that really gets opened is on has HSM control, the I/O channel used is reset to the original device when the file is closed. Such a structure is useful, for example, where a file is moved to a disk jukebox and it is desired to have it remain there indefinitely, but where it is desired to have the file appear to be on some other disk because users or applications have come to expect it there.
For this menu item, you will be asked for the file to link to. If it exists already, it is used intact. Otherwise, the original file is backed up to the desired destination file. Then in either case, if the file is exactly the same size as the original file in bytes, the original file is truncated to zero blocks and a softlink mark is inserted.

The effect if the file didn't previously exist is to move the file to a new location so it is used there for all activities, and to free the space on the original disk. If a link is being set up to a different file somewhere, the effect is just to set that up, leaving the original file alone unless it appears to be a copy of the one linked from.

Note that softlinks MUST reside on local disks (or devices like jukebox magneto optical disks that act like local disks, with software like Virtual Branches which makes them appear online at all times or like Squash which will allow compressed storage but will act like a normal disk).

2. You can set up a "R/O Softlink". This is a cross between normal shelving and softlinks. You are asked where the file should be stored, and the file is moved there. However, the marking set on it causes it to act like a soft link if the file is opened for read only, and like an unshelving if the file is opened for write of any kind. That is, if you open the file for read, it is opened on the device where it is saved, directly and without any intervention by the HSM server. If the file is opened for read/write, however, it is treated as having been shelved and is brought back to normal storage. This makes the "R/O Softlink" ideal for use where files are stored on write-once media or where the linked copy is desired to be archival and untouched. Once the file is unshelved, it is accessed thereafter on normal disk. It should of course be realized that when a file is unshelved in the case of a R/O Softlink, anyone who has the original version open for read only still is accessing the original version of the file and will not see anything done to an unshelved copy. Most situations won't have this issue, but it should be understood so that it will not cause trouble. In either form of softlink, the advantage is that file access is extremely fast where the file need not actually be moved.

3. You can mark the file and move it to a backing storage area. This is normal file shelving. This is done by backing the file up to a save area and marking it as shelved, so that a subsequent open will be able to unshelve it. Note that with this choice, if the logical name `GCY$ZIPHSC` is defined as "`YES`", the storage is in ZIP compressed form. This will save considerable space, but at the expense of significant CPU time during shelving and unshelving. On slower VAXen this time may be prohibitive. On AXP ma-

chines it may however be acceptable. The logical must be set for the unshelving to work properly also.

Note: The shelving and unshelving are actually done under control of the FILSAV.COM and FILRST.COM command files, which can be edited if you feel comfortable doing your own support of those components. (We cannot support custom versions of these files, so if you edit them, you must support the versions you create.)

The last menu item, "Move and mark all selected files now", tells the system to perform the moving of all files you have selected. The other menu selections perform their operations immediately, save for the "`Quit...`" operation which lets you not touch a file. If you select that, it will move to the next file you have selected, if you selected more than one in the fullscreen interface or if you selected a wildcard specification with more than one file matching it. Thus you can mark files one at a time or can mark all at once. Note too that only normal shelving makes sense to do all at once, since softlinks need to have their destinations set individually.

The "Perform database maintenance commands" menu looks like this:

```
          Hierarchical Storage Manager
               MAINTENANCE COMMANDS

     --> Delete this entry
         Remove all database entries for now-deleted files
         Create listing of all files marked
         Repair deleted ACEs
         Change or delete a string in all lists
         Done with this menu
```

This menu allows you to work on markings of files.

`Delete this entry`
means that you are removing markings from the file. Note that this is NOT the same as unshelving a shelved file. Rather, it just removes the marking that would allow it to be unshelved. Where you have made a soft link to a

file, this removes the softlink marking so that if there's another file that was linked to, an open will now get the original file again. (If of course the original file was truncated to zero blocks, this may not be very useful.)

`Remove all database entries for now-deleted files`
searches the database for the current device and ensures that the files it points to still exist. If they do not, the database entries for them are removed. This cleanup is one that should be done periodically. Under normal conditions, there will be times when files are deleted and you will want to clean up the HSM server's database. Note that deleting a shelved file does NOT delete the shelved copy, so the shelved copy remains as a backup of the file.

`Create listing of all files marked`
creates a file in HSD$DB with file IDs of all files in the database so that it can be used for kernel-marking all files to allow unshelving operations to succeed even if markings are disturbed. If you are using the option of having HSM ignore all files except those marked, this allows you to produce current markings as well. The filename will be the same as the HSM database with _LIST appended to it (which encodes device name in the filename).

`Repair deleted ACEs`
ensures that all file markings match those in the HSM database so that any disturbed markings are restored. Any markings that were damaged are reported on your terminal at this time also.

`Change or delete a string in all lists`
allows global replacement of some string in all the database. It is likely this will VERY rarely be useful for anything but can be used where something unusual like a device physical name change occurs. You are prompted for the original string and replacement string.

USAGE CONDITIONS

HSM should be installed and used on all cluster nodes where a VMScluster is in use, and logical name pointers for HSD$DB which point to the device database files must agree. Any node not having HSM installed will not be able to use its benefits, and shelved files will just appear as zero length files.

It should be noted too that HSM resets the end of file pointers on files shelved to the original file size, though the allocated area is correctly left as zero blocks after shelving. If you run the `ANALYZE/DISK` utility, it will complain about this. It is actually a totally harmless condition and will mess nothing up. If this length is reset by `ANALYZE/DISK/REPAIR`, the unshelving process will run **exactly** as before, and the only effect will be that the `DIRECTORY` command will be unable to show the file size. HSM makes no attempt to patch the DIRectory image or other such images since these are not all methods for finding file information. Rather, we use a harmless method of preserving display information to keep file sizes available when a file is outswapped. On inswap, the full and exact information is of course restored. There are unscrupulous outfits which will attempt to convince you that having the end of file pointers set in this way is a form of disk corruption. This is not true; in fact it will cause no harm whatever, apart from generating some warnings if ANALYZE/DISK is performed. Just understand that these numbers are left in this way as a convenience. (If this bothers you it is easy to edit the FILSAV and FILRST command files not to do it.)

Files shelved to a backup area have names changed so that the original file ID is first, and then have the first part of the original file name. If the HSM system must be down, this makes it fairly straightforward to locate shelved files in nearline storage by hand, or to locate such files should they be backed up separately and should someone want to access them from such backups.

Finally, note that finding free space can take considerable time on a large disk, and that unshelving requires a copy of the file back to its original location. This, too, can require considerable time. If you have files which must be opened quickly, ensure that your site policy inhibits them from shelving, and if you want to be sure space won't run out, run the `MAKSPC` script at low load hours so that the time required to shelve inactive files will occur at hours when it is unlikely to cause trouble. You are of course free to tailor the commands in the `MAKSPC` script (and you may want to arrange that extremely large files are not autoshelved, for example). The FIND utility is capable of this sort of thing readily; its help is supplied to make this sort of site configuration possible.

You have the ability, by default, to inhibit shelving of a file (in VMS V6 and later) by setting the file not shelvable, and a site can set images up with the ability to disregard shelving, or can use the provided HSEXEMPT image to exempt a process from HSM activity on a disk. Therefore at both site and individual levels, considerable control of the actions of HSM is provided. Also, individuals can (if the site policy allows) run the HSAUTHMAINT utility via the MOVEFILE or MOVEHSM commands to shelve any files to which they have control in ways they prefer. Beyond this, the site policy governs what is shelved, where, and how, and it is vital that it be set up to provide the kind of storage migration policy that will migrate files as needed.

Remember: you choose your site policy and you choose which disks it applies to. A little advance planning can be a big help in getting a policy that will cause no grief later.