

Re: Mounting identically labelled disks (OVMS73)

Source: <http://unix.derkeiler.com/Newsgroups/comp.os.vms/2004-01/1454.html>

From: News Reader (*news.reader_at_hostname.invalid*)

Date: 01/16/04

Date: Fri, 16 Jan 2004 13:43:33 +0000 (UTC)

On Thu, 15 Jan 2004 23:11:45 -0500, JF wrote:

*>mount one disk without the /SYSTEM or /CLUSTER parameter. It will be
>mounted privately. You can then SET VOLUME/LABEL=newlabel to make the
>drive look different. Unmount and you can remount the drive/SYSTEM at
>which point it should become available.*

THANKS JF, appreciate the help and the set volume /label worked however looks like more is needed to stop VMS's nannying; letting me share nfs share (originally) identically labelled disks.

More solutions please! – The story so far follows...

\$! the original, still on-line

\$ show dev dad5

```
Device Device Error Volume Free Trans Mnt
Name Status Count Label Blocks Count Cnt
DAD5: Mounted 0 REGNUM_DISK2 831432 124 1
```

\$! the backup

\$ show dev dad20

```
Device Device Error Volume Free Trans Mnt
Name Status Count Label Blocks Count Cnt
DAD20: Mounted 0 REGNUM_DISK2 861205 1 1
```

\$! change the label of the privately mounted backup disk and remount /sys

\$ set volume /label:REGNUM_BACK2 dad20:

\$ umount DAD\$REGNUM_BACKUP2

\$ bind /system /write /connect REGNUM_BACKUP2

%LADCP-I-CONNAVAIL, successful connection to service REGNUM_BACKUP2

%LADCP-I-BIND, service bound to logical unit DAD\$REGNUM_BACKUP2 (_DAD22:)

\$ mount /noassist /system /write DAD\$REGNUM_BACKUP2 REGNUM_BACK2

%MOUNT-I-MOUNTED, REGNUM_BACK2 mounted on _DAD22:

\$!

\$! the use of /write is to avoid the possibility of problems we're

\$! seeing being caused by attempts to NFS share a read-only volume

comp.os.vms: Re: Mounting identically labelled disks (OVMS73)

```
$ !
$ ! the disk is an NFS Unix container disk; the logicals and symbols
$ ! are there to get (volatile) physical device and unit numbers out
$ ! of the tcpip commands so that this whole thing can be scripted
$ !
$ define /system /executive /translation:concealed disk_backup_home -
_ $ disk$regnum_back2:[000000.]
$ show log disk_backup_home
  "DISK_BACKUP_HOME" = "DISK$REGNUM_BACK2:[000000.]" (LNM$SYSTEM_TABLE)
$
$ translation = f$string(f$trnlm("disk_backup_home") - ":[000000.]")
$ show symb translation
  TRANSLATION = "DISK$REGNUM_BACK2"
$
$ tcpip map "/disk_backup_home" 'translation
$ !
$ ! succeeded; underlying volume has been mapped. Now the container...
$ !
$ translation = f$string(f$trnlm("disk_backup_home") - ":[000000.]" + -
_ $ ":[export]")
$ show symb translation
  TRANSLATION = "DISK$REGNUM_BACK2:[export]"
$
$ tcpip map "/backup" 'translation
%TCPIP-E-MAPERROR, error processing MAP or UNMAP request
-SYSTEM-F-INCVOLLABEL, incorrect volume label
$ !
$ ! failed; the container DISK$REGNUM_BACK2:[export] has not been
$ ! mapped to a Unix share called /backup
$ !
$ ! Section 10.10.6 of the TCPIP V5.1 Management manual states
$ ! "If you are making copies or restoring files from a backup.
$ ! The container file records the volume label and the Files--11
$ ! file identifiers of the actual files on the disk. If you copy
$ ! the file system or change the volume label, you must run ANALYZE
$ ! CONTAINER/REPAIR after you copy the files so that the file
$ ! identifiers and volume label are corrected for the new location of
$ ! the files."
$ !
$ ! Right, so since we've changed the label and the container has
$ ! recorded the old label we need to fix this up with the /repair
$ ! option. Here goes...
$ !
$ tcpip analyze container /repair 'translation
%TCPIP-I-ANA_STARTSUPER, starting super block validation
%%%%%%%% OPCOM 16-JAN-2004 12:50:27.15 %%%%%%%%%
Message from user SYSTEM on VMSBOX
%TCPIP-E-UFS_BAD_ID, container file system volume identification does not
match OpenVMS disk volume
$ !
$ ! Yeah, that's why I've asked you to fix it up, so what gives?
```

comp.os.vms: Re: Mounting identically labelled disks (OVMS73)

\$!