Getting Started with DeviceShare

Order Number: AD-200-DEV-GS/2.0

DeviceShare provides facilities for remote sharing of tape and/or disk devices throughout the network. This manual describes the concepts and features of DeviceShare.

Revision/Update Information: This is a new manual

Operating System and Version: VAX/VMS Version 5.2 or later

Software Version: DeviceShare Version 2.0

Advanced Systems Concepts
Hoboken, New Jersey

First Printed: February 1992

The information in this document is subject to change without notice and should not be construed as a commitment by Advanced Systems Concepts, Inc. ASCI assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license, and may only be used or copied in accordance with the terms of such license.

ASCI, ASCI logo, RSO, PSS, EPSS, INTACT, LIBERATOR, PRISM, RECORDER, RQE, SHADOW, DEVICESHARE, TAPESHARE, VIRTUOSO, and WATCH are trademarks of Advanced Systems Concepts, Inc.

DIGITAL, VAX, and VMS are trademarks of Digital Equipment Corporation.

Copyright ©1992 Advanced Systems Concepts, Inc., Hoboken, New Jersey 07030

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without the written permission of Advanced Systems Concepts, Inc.

Contents

Preface

The DeviceShare software product, engineered by Advanced Systems Concepts, Inc., allows you to share your mass storage devices among all or selected nodes of your network or VAXcluster. This manual is designed to help you to quickly integrate DeviceShare into your system. It explains the terms, concepts, and procedures you will need to install and use the product within a minimum amount of time. For a complete, in-depth description of the product, refer to the *DeviceShare Guide to Operations* manual.

1 Overview

The DeviceShare software product extends the usefulness and accessibility of your VAX mass storage devices by making them available to any or all of the nodes within your DECnet network and VAXcluster. Use of DeviceShare eliminates the need of providing each of your VAX systems with expensive disk and tape drives and facilitates such applications as remote backup, data sharing, and remote software installation.

DeviceShare uses the Client/Server mode of computing. Each of the systems on which DeviceShare is installed may be defined as either a Server or Client node. A *Server* node physically possesses one or more mass storage devices and provides their services to designated *Client* nodes. A Server can also act as a Client if it makes use of a device served by another node.

This chapter discusses several of the features provided by DeviceShare.

Note: For the sake of clarity, each of the VAX nodes, presented in examples of DCL syntax, will prompt with its node name followed by a *$*. For example, the DCL prompt for VAX node OMEGA will be *OMEGA$*.

1.1 Network Access

DeviceShare serves tapes and disks across a network, allowing greater resource utilization of equipment than was previously possible.

1.1.1 Serving Tape Devices

Node OMEGA is a VAX 4000 Model 300 that has a local 6250 bpi tape drive, named 1MUA0, that is used for high-speed disk backups. Through the use of a simple DeviceShare command, OMEGA can be instructed to share the tape drive with other nodes in the network:

```
OMEGA$ DEVICESHARE SERVE $1$MUA0
```

When this command is issued, any Client node may request use of OMEGA's tape drive.

Node EPSILON is a VAXstation 3100 that needs to back up its disks. When one of EPSILON's users enters the following command:

```
EPSILON$ DEVICESHARE ALLOCATE OMEGA::$1$MUA0 BACKUP_DEVICE
```

the DeviceShare client software on EPSILON will be activated to establish a logical link with a DeviceShare server on OMEGA. Once the link has been successfully established, OMEGA's tape drive will appear to be a local device on EPSILON. As specified in the ALLOCATE command, the user has chosen to assign the system-wide logical name *BACKUP_DEVICE* to the tape drive.

EPSILON's users may now perform any valid I/O operation to the tape device, just as though it was physically attached to their system. Whenever a reference must be made to the tape drive, the logical name BACKUP_DEVICE may be used.

```
EPSILON$ INITIALIZE BACKUP_DEVICE SEP92
EPSILON$ MOUNT/FOREIGN BACKUP_DEVICE
EPSILON$ BACKUP/LOG $1$DUA1: BACKUP_DEVICE:10SEP92.BCK/SAVE_SET
EPSILON$ DISMOUNT BACKUP_DEVICE
```

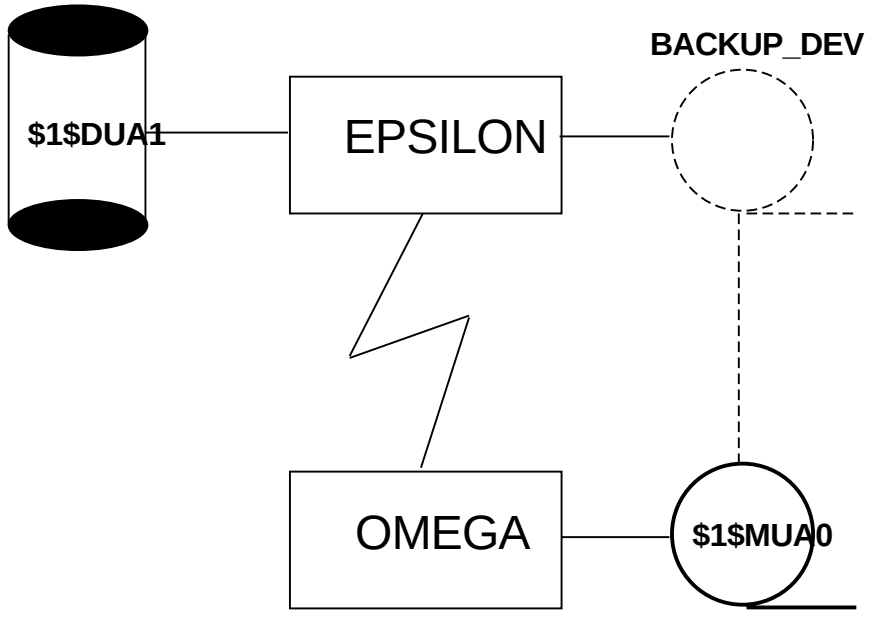
This small sequence of DCL commands first initializes the tape volume on the remote device, labelling it *SEP92*. The tape is then mounted locally as a FOREIGN device and a backup of the local disk, 1DUA1, is performed to it. When the backup completes, the tape is dismounted.

When EPSILON is finished using the tape drive, its control may be relinquished by typing

```
EPSILON$ DEVICESHARE DEALLOCATE BACKUP_DEVICE
```

At that point, the tape drive will no longer be available to EPSILON and the DECnet link between OMEGA and EPSILON will be disconnected.

Figure 1–1 OMEGA Serving Tape to EPSILON



1.1.2 Serving Disk Devices

By their very nature, magnetic tape drives are sequential devices. While a tape drive is in use by a Client node, it cannot be used by another node. However, DeviceShare is not limited to the support of tape devices. Disk devices can also be shared using DeviceShare.

An alternative approach to the above problem would be to reverse the roles of OMEGA and EPSILON. EPSILON could be designated as the Server node to make its disk available for network access. On node EPSILON, the following command:

```
EPSILON$ DEVICESHARE SERVE $1$DUA1
```

allows sharing of the RA81 disk, named 1DUA1. For node OMEGA to gain access to this disk, a DeviceShare ALLOCATE command must be issued:

```
OMEGA$ DEVICESHARE ALLOCATE EPSILON::$1$DUA1 SOURCE_DISK/SHARE/SYSTEM
```

On the successful completion of this command, EPSILON's disk will appear to be a local RA81 on OMEGA, and can be accessed using the logical name *SOURCE_DISK*. Note the use of the */SHARE* qualifier. The presence of this qualifier tells DeviceShare to allocate the disk to OMEGA for *non-exclusive access*. This means that the disk can be accessed by both nodes. If the /SHARE qualifier was omitted, and 1DUA1 was not currently mounted on EPSILON, only OMEGA would be able to use the disk.

The */SYSTEM* qualifier indicates that EPSILON's disk is to be available on a system-wide basis to OMEGA. DeviceShare will mount the disk as a public volume on OMEGA.

The disk backup can now proceed as before, with OMEGA assuming the role of Client node:

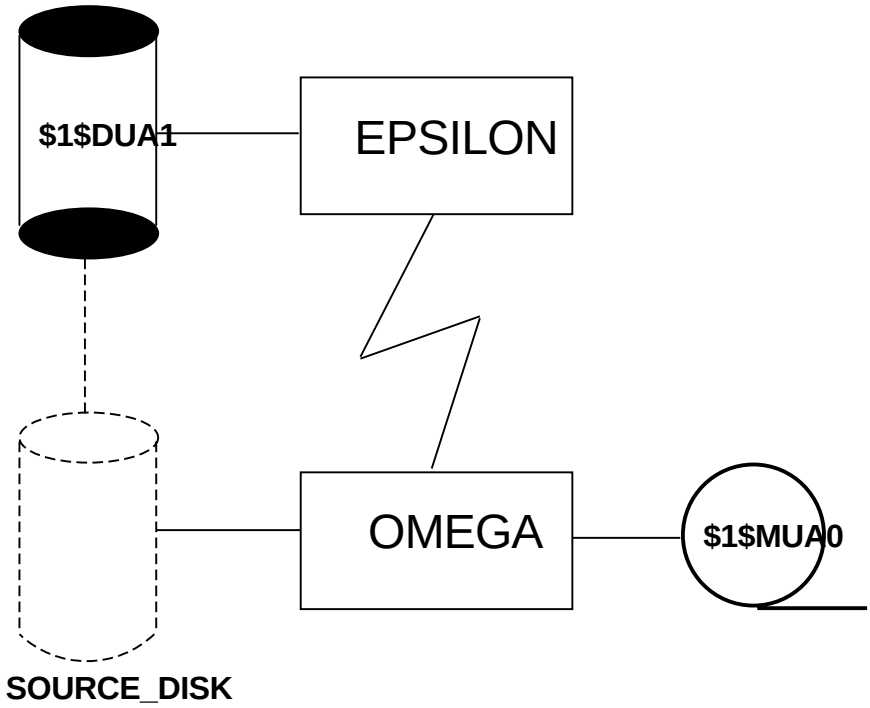
```
OMEGA$ INITIALIZE $1$MUA0 SEP92
OMEGA$ MOUNT/FOREIGN $1$MUA0
OMEGA$ BACKUP/LOG SOURCE_DISK: $1$MUA0:10SEP92.BCK/SAVE_SET
```

This command stream performs the same function as the first example, except that EPSILON's disk is mounted locally on OMEGA. The disk is still mounted on and in use by EPSILON.

When OMEGA's operator has completed the backup of EPSILON's disk, access to the disk can be removed by issuing a DeviceShare DEALLOCATE command.

```
OMEGA$ DEVICESHARE DEALLOCATE SOURCE_DISK
```

Figure 1–2 EPSILON serving disk to OMEGA



As you can see from the preceding examples, once a device has been allocated by DeviceShare, use of that device is totally transparent to the client. Standard DCL commands and VMS services are used to access the device.

1.2 VAXcluster Tape Support

In addition to network support, DeviceShare allows the sharing of magnetic tape devices between nodes of a VAXcluster. In this mode, DeviceShare uses standard SCS services and the MSCP protocol.

To use this method of sharing, the */SCS* qualifier must be supplied on the DeviceShare SERVE command:

```
OMEGA$ DEVICESHARE SERVE MUA0/SCS
```

Once this command has been issued, all nodes in the VAXcluster will have immediate access to OMEGA's tape drive.

Use of this method is limited to magnetic tape devices and to nodes within the same VAXcluser. DeviceShare allows a tape drive to be served by either or both methods.

1.3 Generic Device Naming

When a Server node makes a local device available for network sharing, one or more *aliases* may be assigned to the device. As the term implies, an alias is simply another way to reference the served device. In the example presented in Section 1.1.1, node OMEGA's system manager may have specified the alias *6250BPI* when serving the tape drive:

```
OMEGA$ DEVICESHARE SERVE $1$MUA0/ALIAS=6250BPI
```

This feature is especially useful when multiple devices of similar types are being served. If all of these devices are assigned the same alias, DeviceShare will select the first free device to satisfy a remote service request. For example, if node OMEGA serves two tape drives:

```
OMEGA$ DEVICESHARE SERVE $1$MUA0/ALIAS=6250BPI
OMEGA$ DEVICESHARE SERVE $1$MUA1/ALIAS=6250BPI
```

a client may request use of a tape via the alias name.

```
EPSILON$ DEVICESHARE ALLOCATE OMEGA::6250BPI BACKUP_DEVICE
```

When OMEGA receives EPSILON's request, it will attempt to satisfy the request with device 1MUA0. If that device is not available, OMEGA will try to allocate 1MUA1. With this facility, clients need not be aware of the specific device names on their servers.

DeviceShare will always create a *built-in* alias of the device type (e.g., TU81, TK50, RA90, etc) for each served device.

1.4 Automatic Network Reconnection

Occasionally, transient network failures may occur while DeviceShare operations are active. If they do, DeviceShare provides an *automatic network restart* feature that will immediately attempt to reconnect the link. By default, DeviceShare will wait up to fifteen minutes for the reconnection to take place. If a successful reconnection is made, all operations to the device will proceed, as before. If not, the current operations will complete with a *Device Not in Configuration* failure. This feature can prevent unnecessary repetition of work, especially if a link failure occurs in the middle of a long, multi-reel backup.

1.5 Automatic Device Serving

DeviceShare provides a startup command procedure that, in addition to starting the system, allows the automatic serving of all or selected disks or tapes. Refer to Section 1.11 for details of this feature.

1.6 Restricting Access to Shared Devices

When a device is served by DeviceShare, it will, by default, be available to all the nodes in the network. However, DeviceShare provides a method of restricting access to the devices it serves.

The */INCLUDE* qualifier of the DeviceShare SERVE command specifies a list of nodes that are authorized to request use of the device.

```
EPSILON$ DEVICESHARE SERVE $1$DUA1/INCLUDE=(OMEGA,GAMMA)
```

With this command, EPSILON allows its disk to be accessed by nodes OMEGA and GAMMA, only. Attempts to access the disk from any other nodes will be rejected.

The obverse of the /INCLUDE qualifier is the */EXCLUDE* qualifier. This qualifier specifies a list of nodes that are *not* authorized access to the device.

```
EPSILON$ DEVICESHARE SERVE $1$DUA1/EXCLUDE=BETA
```

This command serves the device to all nodes in the network but BETA. Any requests for the disk received from the BETA node will be rejected.

For complete details on the security features offered by DeviceShare, refer to the *DeviceShare Guide to Operations* manual.

1.7 Network Data Compression

DeviceShare can compress repetitive strings of characters that are sent across a network link. Depending on the characteristics of your data, this feature can significantly increase the throughput and speed of your remote operations. To activate this feature, specify the */COMPRESS* qualifier on the DeviceShare SERVE command:

```
OMEGA$ DEVICESHARE SERVE $1$MUA0/COMPRESS
```

Please note that data is compressed only while it is being transmitted over the network. It is not written to the served device in compressed form.

Specifying compression at the Server end means that compression will be in effect for all Clients that allocate the device. Compression can also be activated on the Client side, regardless of the setting at the Server. This is accomplished by supplying the /COMPRESS qualifier on the DeviceShare ALLOCATE command. When activated in this manner, compression will only be in effect for the specific client's session.

1.8 Displaying Served-Device Statistics

DeviceShare accumulates various statistics about each device it serves. To display these statistics, issue the DeviceShare SHOW DEVICE command.

```
OMEGA$ DEVICESHARE SHOW DEVICE MUA0

DeviceShare V020-003 Shared Devices on OMEGA 24-JAN-1992 16:01:40.02
Copyright (C) 1992 by Advanced Systems Concepts, Inc.  All Rights Reserved.
Licensed to ASCI-INTERNAL

Device: MUA0                        Status: Available    Network-Served
  ----------I/O Counts----------   --Reads---   --Writes--
  I/O Operations               :        1027            0
  Characters Transferred       :     1670548            0
```

If the device name is omitted from the SHOW DEVICE command, statistics on all served devices will be displayed. Additional information is displayed for devices that have the compression feature enabled.

```
OMEGA$ DEVICESHARE SHOW DEVICE MUA1

DeviceShare V020-003 Shared Devices on OMEGA 24-JAN-1992 16:33:43.10
Copyright (C) 1992 by Advanced Systems Concepts, Inc.  All Rights Reserved.
Licensed to ASCI-INTERNAL

Device: $1$MUA1                    Status: In Use       Network-Served
  ----------I/O Counts----------   --Reads---   --Writes--
  I/O Operations               :       12910        13172
  Characters Transferred       :    49140726     32616692
  Compressed Character Counts  :    10097770      6639129
  Compression Effectiveness    :  %       20    %      20
```

1.9 Remote Operator Communications

DeviceShare's users can send messages to and receive replies from operators on remote systems, in a manner similar to that of the VMS REPLY command. These messages can ask an operator to perform a specific action and, optionally, request a reply. The remote operator's reply can be placed into a DCL symbol for program or command procedure analysis.

Assume that an EPSILON operator must restore a disk from a backup tape. Since the tape drive is physically located on node OMEGA, the operator of that system must be told which tape to mount. To do this, the EPSILON operator executes the following command procedure:

```
$   DEVICESHARE ALLOCATE OMEGA::6250BPI RESTR_TAPE
$   DEVICESHARE OPCOM/REPLY=ANSWER RESTR_TAPE "Please mount tape SEP92"
$   IF ANSWER .NES. "OK" THEN GOTO DEALLOCATE_TAPE
$   MOUNT/FOREIGN RESTR_TAPE
$   BACKUP/IMAGE RESTR_TAPE:10SEP92.BCK/SAVE $1$DUA1
$   DISMOUNT RESTR_TAPE
$   DEVICESHARE OPCOM RESTR_TAPE "Please remove tape SEP92"
$ DEALLOCATE_TAPE:
$   DEVICESHARE DEALLOCATE RESTR_TAPE
$   EXIT
```

When the command procedure is run, the "Please mount tape SEP92" message will appear on the OMEGA operator's console. Note the use of the REPLY qualifier on the first OPCOM command. This qualifier indicates that a reply to the mount message is expected from OMEGA's operator. When it is received, it will be placed into the DCL symbol ANSWER. This symbol need not exist before the OPCOM command is issued, but, if it does, it must be a string datatype. If OMEGA's reply is "OK", the procedure proceeds to mount the tape locally and restore the disk. When the restore has completed, the procedure again sends a message to OMEGA, asking the operator to physically remove the tape from the drive. This message requires no reply.

1.10 Product Installation

Prior to its use, DeviceShare must be installed on all the Server and Client nodes that will have need of it. Installation is a quick and simple process that uses the VMSINSTAL and LMF facilities provided by VMS - just like any standard Digital software product you may be using. The installation procedure is described in Chapter 2.

1.11 Starting DeviceShare

As part of installation, a command procedure named DEVICESHARE_STARTUP.COM is placed in the SYS$STARTUP directory. This procedure is used to start up the product and it must be executed prior to issuing any DeviceShare commands.

When you run the startup procedure, you can supply up to four parameters. The first parameter simply informs DeviceShare whether the node it will be running on is a Server or Client. If this parameter is omitted, a Server node will be assumed. Note that a Server node provides both Server *and* Client functions.

```
OMEGA$ @SYS$STARTUP:DEVICESHARE_STARTUP SERVER
```

The second parameter tells DeviceShare which transport methods to use: NETWORK, SCS, or BOTH. If you omit this parameter, the NETWORK method will be used.

```
OMEGA$ @SYS$STARTUP:DEVICESHARE_STARTUP SERVER NETWORK
```

The third parameter performs *automatic device sharing*, as referred to in Section 1.5. You may specify TAPE, DISK, or BOTH. If you choose TAPE, all tape devices on the system will be served automatically. Similarly, if you choose DISK, all your disk devices will be served. If you choose BOTH, all the mass storage devices on your system will be served. If the parameter is omitted, no devices will be automatically served.

```
OMEGA$ @SYS$STARTUP:DEVICESHARE_STARTUP SERVER NETWORK TAPE
```

If you specified automatic device serving, the fourth parameter enables you to supply additional qualifiers to the SERVE commands that will be issued. For example, if you want node OMEGA to serve all its tape devices to nodes BETA and EPSILON, but not to any other nodes, execute the startup procedure as follows:

```
OMEGA$ @SYS$STARTUP:DEVICESHARE_STARTUP SERVER NETWORK TAPE -
       "/INCLUDE=(BETA,EPSILON)"
```

2 Installation

Installation of DeviceShare is quite simple. It consists entirely of standard Digital installation procedures and takes only minutes to perform. This chapter provides an overview of the installion procedure. For complete details on product installation, refer to the *DeviceShare Guide to Operations* manual.

2.1 Contents of the Distribution Kit

To determine if your distribution kit is complete, compare the contents with the following list:

DeviceShare Product Checklist

- *Getting Started with DeviceShare*™ Manual
- *DeviceShare*™ *Guide to Operations* Manual
- DeviceShare™ Distribution Media
- DeviceShare™ Product Information Sheet
- DeviceShare™ Software Product License Agreement
- DeviceShare™ Product Authorization Key (PAK)

If the contents of this kit are incomplete, contact Advanced Systems Concepts, Inc., Hoboken, NJ, or your local distributor.

The Product Authorization Key (PAK) may be delivered separately.

2.2 Product Authorization Key

DeviceShare uses Digital's License Management Facility (LMF) to determine software license compliance. LMF requires that a Product Authorization Key (PAK) be registered. You will receive a DeviceShare PAK as part of your installation kit. Though not strictly required, ASCI strongly recommends that you register the PAK before installing DeviceShare.

To register a PAK, log into a privileged account and execute the VMSLICENSE.COM command procedure that is located in the SYS$UPDATE directory. The following example shows a sample run of VMSLICENSE, registering a PAK for a DeviceShare SERVER license. For complete information on the use of this procedure, and on LMF in general, see the *VMS License Management Facility Manual*, available from Digital Equipment Corporation.

Example 2–1 Sample PAK Entry Using VMSLICENSE

```
$ @SYS$UPDATE:VMSLICENSE RETURN

    VMS License Management Utility Options:

        1. Register a Product Authorization Key
        2. Amend an existing Product Authorization Key
        3. Cancel an existing Product Authorization Key
        4. List Product Authorization Keys
        5. Modify an existing Product Authorization Key

        9. Exit this procedure

    Type '?' at any prompt for a description of the information
    requested.

Enter one of the above choices [1] 1 RETURN
Do you have your Product Authorization Key? [YES] YES RETURN

   The REGISTER option allows you add a new license to a license
   database.  A Product Authorization Key (PAK) provides the product
   name and information you need to register the license.  You must
   enter all the information provided by your PAK exactly as specified.


PAK ID:
                      Issuer [DEC]ASCI RETURN
         Authorization Number []ASCI-92001-001 RETURN

PRODUCT ID:
                Product Name []DEVICESHARE RETURN
                    Producer [DEC]ASCI RETURN

NUMBER OF UNITS:
              Number of Units []0 RETURN

KEY LEVEL:
                       Version [] RETURN
         Product Release Date [] RETURN

KEY TERMINATION DATE:
         Key Termination Date []12-APR-1990 RETURN

RATING:
      Availability Table Code []F RETURN
          Activity Table Code [] RETURN

MISCELLANEOUS:
                  Key Options []MOD_UNITS RETURN
                Product Token []XASCI-SAMPLE RETURN
                  Hardware-Id []XASCI-SAMPLE RETURN
                     Checksum []4-AAAA-BBBB-CCCC-DDDD RETURN
```

Example 2–1 Cont'd on next page

Example 2–1 (Cont.) Sample PAK Entry Using VMSLICENSE

```
      License Database File:  SYS$COMMON:[SYSEXE]LMF$LICENSE.LDB
                     Issuer:  ASCI
              Authorization:  ASCI-92001-001
                   Producer:  ASCI
               Product Name:  DEVICESHARE
                      Units:  0
                       Date:
                    Version:
           Termination Date:  12-APR-1990
               Availability:  F
                   Activity:
                    Options:  MOD_UNITS
                      Token:  XASCI-SAMPLE
                Hardware ID:  XASCI-SAMPLE
                   Checksum:  4-AAAA-BBBB-CCCC-DDDD

Is this information correct? [YES] YES RETURN
Registering DEVICESHARE license in SYS$COMMON:[SYSEXE]LMF$LICENSE.LDB...
```

2.3 Installing DeviceShare

To install DeviceShare, log into a privileged account, set the default to SYS$UPDATE:, and then invoke VMSINSTAL, the standard VMS software installation command procedure. The example that follows presents a DeviceShare SERVER installation. For complete details on the installation process, refer to the *DeviceShare Guide to Operations* manual.

Example 2–2 A Sample Installation

```
$ @SYS$UPDATE:VMSINSTAL RETURN

 VAX/VMS Software Product Installation Procedure V5.4-1


It is 23-DEC-1991 at 09:27.

Enter a question mark (?) at any time for help.
* Are you satisfied with the backup of your system disk [YES]? YES RETURN
* Where will the distribution volumes be mounted: $2$MUA0: RETURN

Enter the products to be processed from the first distribution volume set.
* Products: DEVSHR020 RETURN
* Enter installation options you wish to use (none): RETURN
The following products will be processed:
  DEVSHR V2.0
   Beginning installation of DEVSHR V2.0 at 09:28

%VMSINSTAL-I-RESTORE, Restoring product save set A ...
%VMSINSTAL-I-RELMOVED, Product's release notes have been moved to SYS$HELP.

    DeviceShare V020-000 Installation Procedure
    Copyright (C) 1990, 1991, Advanced Systems Concepts, Inc.

******************************************************************************

Attention -  System Manager / Installer:

1.      Please remember to add SYS$STARTUP:DEVICESHARE_STARTUP
        to your VMS Startup procedures.

2.      DeviceShare Release Notes can be found in SYS$HELP.

3.      DeviceShare DCL commands will be added to your DCL Tables.

4.      DeviceShare Help will be added to your VMS Help library.

5.      This product requires an ASCI Product Authorization Key (PAK)
        which is registered using Digital's License Management System.
        While you will be able to install the product without a PAK,
        you must register the license prior to using the product.

******************************************************************************

* Do you want to purge files replaced by this installation [YES]?

DeviceShare requires a location for its files.

* Please enter device/directory for DeviceShare [SYS$SYSDEVICE:[DEVICESHARE]]:
SYS:[DEVSHR020] RETURN
%VMSINSTAL-I-SYSDIR, This product creates system disk directory  SYS$SYSDEVICE:[DEVSHR020].
* Is this a CLIENT-only installation [NO]? NO RETURN

          Product:      DEVICESHARE
          Producer:     ASCI
          Version:      2.0
          Release Date: 1-DEC-1991
```

Example 2–2 Cont'd on next page

Example 2–2 (Cont.) A Sample Installation

```
* Does this product have an authorization key registered and loaded? YES RETURN

+------------------------------------------------------------------------+
!                                                                        !
! Network Device Serving use a UAF-account, and a DECnet object named    !
! DEVSHR_NTS.                                                            !
!                                                                        !
! This portion of the installation creates the UAF-record,               !
! and defines/sets the DECnet object.                                    !
!                                                                        !
+------------------------------------------------------------------------+
!                                                                        !
! In order to insure that NTS runs with the proper quotas                !
! and privileges, the DEVSHR_NTS account will be created.                !
!                                                                        !
! You may modify the attributes of this account after the installation   !
! is complete to meet your site needs, provided you do not lower any     !
! quotas or remove privileges.                                           !
!                                                                        !
+------------------------------------------------------------------------+
* Enter the PASSWORD for DEVSHR_NTS account (minimum 12 characters):

* Verify:

* Enter the UIC (include brackets) [[3342,3342]]: RETURN

* Enter the DEVICE where the DEVSHR_NTS directory will reside [SYS$SYSDEVICE]: RETURN
%VMSINSTAL-I-ACCOUNT, This installation creates an ACCOUNT named DEVSHR_NTS.
%UAF-I-ADDMSG, user record successfully added
%UAF-I-RDBADDMSGU, identifier DEVSHR_NTS value: [003342,003342] added to rights data base
%UAF-I-RDBADDMSGU, identifier DEVSHR value: [003342,177777] added to rights data base
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DEVSHR_NTS.
%UAF-I-MDFYMSG, user record(s) updated
%DEVSHR-I-CREDIR, Creating SYS$SYSDEVICE:[DEVSHR_NTS] directory
%VMSINSTAL-I-SYSDIR, This product creates system disk directory  SYS$SYSDEVICE:[DEVSHR_NTS].
%DEVSHR-I-MODUAF, Modifying account
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DEVSHR_NTS.
%UAF-I-MDFYMSG, user record(s) updated
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DEVSHR_NTS.
%UAF-I-MDFYMSG, user record(s) updated
%VMSINSTAL-I-ACCOUNT, This installation updates an ACCOUNT named DEVSHR_NTS.
%UAF-I-MDFYMSG, user record(s) updated
%DEVSHR-I-DEFNTSREM, Defining DEVSHR_NTS object in DECnet database
%DEVSHR-I-SETNTSREM, Setting DEVSHR_NTS object in DECnet database
%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...

 Installation of DEVSHR V2.0 completed at 09:33

Enter the products to be processed from the next distribution volume set.
* Products: exit RETURN
 VMSINSTAL procedure done at 09:34
```

3 Using DeviceShare

DeviceShare can solve many problems concerning remote device accessibility. This chapter discusses several common scenarios in which DeviceShare can be employed to make life easier. In the discussions that follow, assume a four-node network, consisting of the following VAX sytems: OMEGA, BETA, EPSILON, and GAMMA. Of the four, only OMEGA possesses a tape drive, but all of the nodes require disk backup.

3.1 Decentralized Backup

One solution to the above problem is to provide a *decentralized backup* facility, in which OMEGA serves its tape to the other nodes. This situation is illustrated in Figure 3–1. Each node can then perform its own backup when OMEGA's tape drive is available.

The first step is for OMEGA to make the tape available to the network:

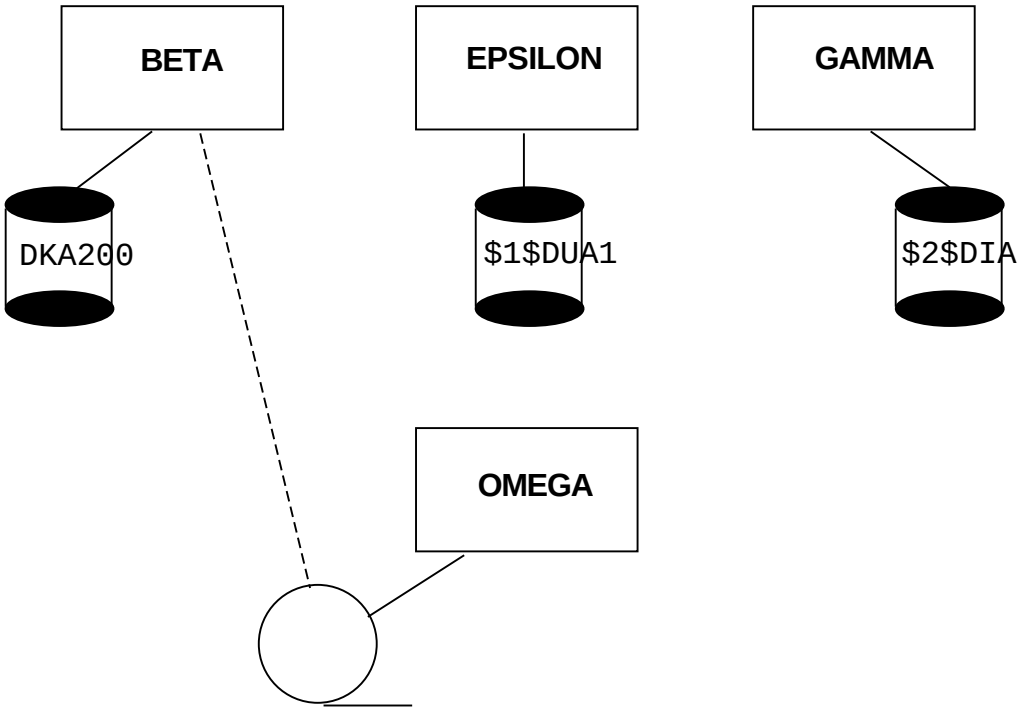
```
OMEGA$ DEVICESHARE SERVE $1$MUA0/ALIAS=6250BPI
```

The other nodes can, in turn, allocate the device, perform their backups, and release the tape for other use. For example, node BETA needs to backup its RZ23 disk, named DKA200. To do so, BETA's operator issues the following commands:

```
BETA$ DEVICESHARE ALLOCATE OMEGA::6250BPI TAPE
BETA$ INITIALIZE TAPE SEP92
BETA$ MOUNT/FOREIGN TAPE
BETA$ BACKUP/LOG/IMAGE DKA200: TAPE:10SEP92.BCK/SAVE_SET
BETA$ DISMOUNT TAPE
BETA$ DEVICESHARE DEALLOCATE TAPE
```

Note that BETA uses the alias name (6250BPI) to allocate OMEGA's tape drive, rather than the actual device name. Similar command streams may be run on EPSILON and GAMMA. Using this method, all nodes can backup their disks, but each must wait its turn for the tape drive to be available.

Figure 3–1 Decentralized Backup



OMEGA serves tape to BETA. BETA performs backup. Othe must wait until BETA completes its backup.

3.2 Centralized Backup

An alternate solution to the problem posed in Section 3.1 is to provide a *centralized backup* facility. In this method, BETA, EPSILON, and GAMMA serve their disks to OMEGA:

```
BETA$ DEVICESHARE SERVE DKA200/INCLUDE=OMEGA
EPSILON$ DEVICESHARE SERVE $1$DUA1/INCLUDE=OMEGA
GAMMA$ DEVICESHARE SERVE $2$DIA3/INCLUDE=OMEGA
```

With the preceding commands, the three tapeless VAX'es have made their disks available for network access by the OMEGA node. OMEGA can now access each of the disks and archive them to its local tape device, while they are still in use on their respective nodes. To aid in this centralized facility, OMEGA's operator executes a command procedure named CENTRAL_BACKUP.COM. This command procedure contains the following statements:

```
$ DEVICESHARE ALLOCATE/SHARE 'P1'::'P2' BACKUP_DISK
$ INITIALIZE $1$MUA0 BACKUP
$ MOUNT/FOREIGN $1$MUA0
$ BACKUP/LOG BACKUP_DISK $1$MUA0:'P1'.BCK/SAVE_SET
$ DISMOUNT $1$MUA0
$ DEVICESHARE DEALLOCATE BACKUP_DISK
```

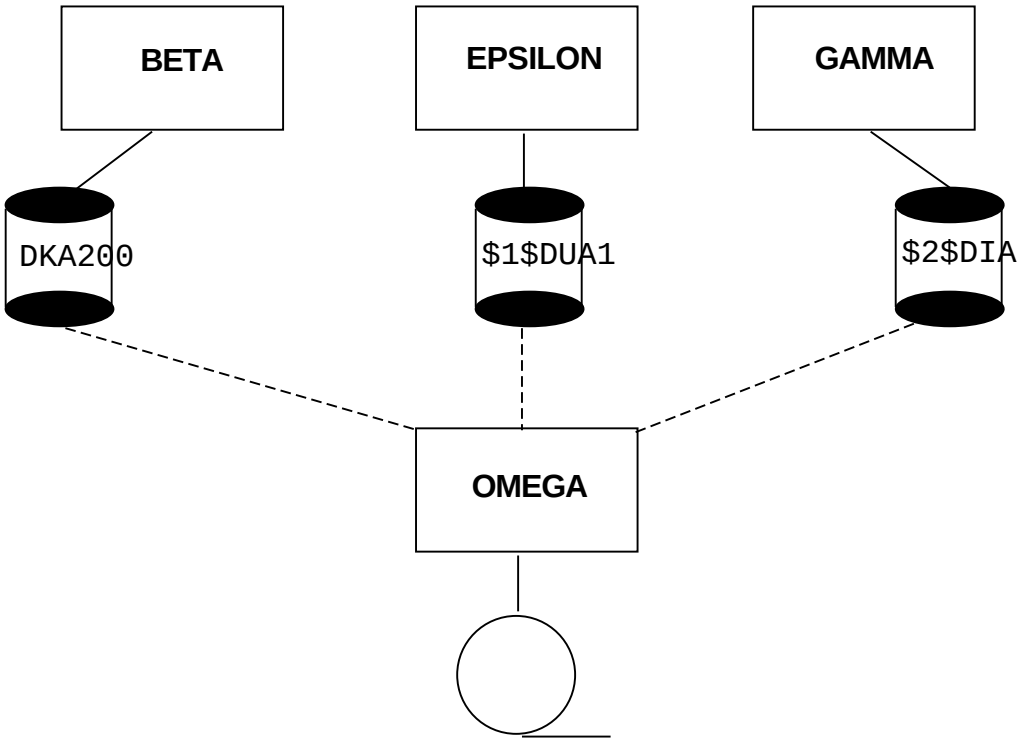
Note that the command procedure has two parameters, which will be supplied by the operator when the procedure is run. The first parameter (P1) is the name of the DECnet node that is serving the disk. The second parameter (P2) is the device name of the disk to be archived. The command procedure performs the following steps:

1. Accesses the remote disk via DeviceShare, making it a local, shared device on OMEGA
2. Initializes the tape on the local drive with the label BACKUP
3. Mounts the local tape as a FOREIGN device
4. Archives the remote node's disk to tape via the VMS BACKUP utility
5. Dismounts the local tape when the Backup has completed
6. Deallocates the disk via DeviceShare

Using this procedure, OMEGA's operator need only issue the following commands to backup the remote nodes' disks:

```
OMEGA$ @CENTRAL_BACKUP BETA DKA200
OMEGA$ @CENTRAL_BACKUP EPSILON $1$DUA1
OMEGA$ @CENTRAL_BACKUP GAMMA $2$DIA3
```

Figure 3–2 Centralized Backup



3.3 Remote Software Installations

New releases of software products, both from Digital and third party vendors, are generally distributed on tape media, either magtape reels or cassettes, such as the TK50. Installing these products on tapeless systems can be inconvenient. Space must be allocated on disk to contain the distribution savesets and those savesets must be copied over the network before installation can take place.

If DeviceShare is installed, a Client VAX can simply allocate a remote tape as a local device and perform the installation directly. For example, if node GAMMA must install a new version of the VAX COBOL compiler, its operator can allocate OMEGA's tape drive, mount the VAX COBOL distribution tape, and run the VMSINSTAL procedure.

```
GAMMA$ DEVICESHARE ALLOCATE OMEGA::6250BPI DISTRIB_TAPE
GAMMA$ @SYS$UPDATE:VMSINSTAL COBOL044 DISTRIB_TAPE OPTIONS N
GAMMA$ DEVICESHARE DEALLOCATE DISTRIB_TAPE
```

Software can also be distributed on Compact Disk (CDROM) diskettes. As with tape devices, DeviceShare can serve CDROM disks to other nodes to ease installation. In our sample four-node network, node EPSILON possesses an RRD40 CDROM disk drive, named 1DUB0. To make the drive available to the network, EPSILON's operator issues a DeviceShare SERVE command:

```
EPSILON$ DEVICESHARE SERVE $1$DUB0/ALIAS=CDROM
```

Notice that the alias *CDROM* was assigned to the device. Node BETA needs to install Version 2.3 of WHIZBANG, a third-party software product that is distributed on CDROM diskettes. To accomplish this, BETA's operator can allocate EPSILON's disk drive, then proceed with the installation, as normal:

```
BETA$ DEVICESHARE ALLOCATE EPSILON::CDROM WHIZ_DISK
BETA$ @SYS$UPDATE:VMSINSTAL WHIZBANG023 WHIZ_DISK OPTIONS N
BETA$ DEVICESHARE DEALLOCATE WHIZ_DISK
```

3.4 Shared Data Access

In many instances, it is necessary to share access to files or databases among multiple users on different nodes. For example, a joint development project may be undertaken by workstation users located at various remote sites. Since these workstations are at different physical locations, clustering might not be feasible. With DeviceShare, all the required data disks could be made to appear as local devices to all the nodes, completely transparent to both users and programs.

First, each node can make its common disks available:

```
BETA$ DEVICESHARE SERVE DKA200

EPSILON$ DEVICESHARE SERVE $1$DUA1

GAMMA$ DEVICESHARE SERVE $2$DIA3
```

Then, each node that requires the remote disks can allocate them and begin to work:

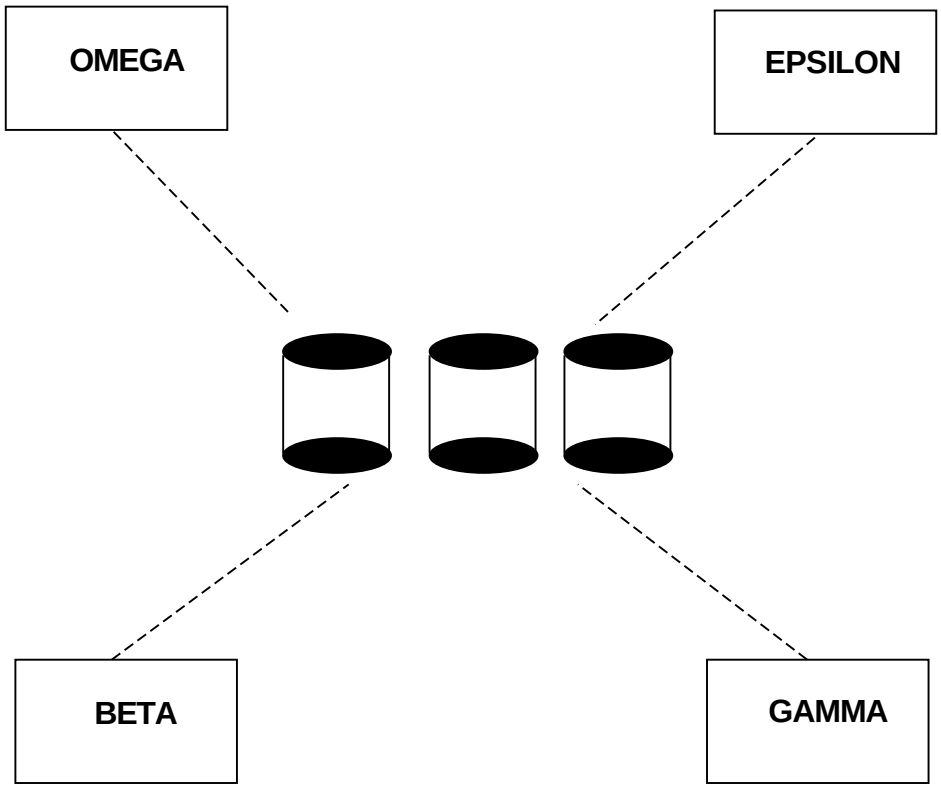
```
OMEGA$ DEVICESHARE ALLOCATE/SHARE BETA::DKA200 DATA_DISK1
OMEGA$ DEVICESHARE ALLOCATE/SHARE EPSILON::$1$DUA1 DATA_DISK2
OMEGA$ DEVICESHARE ALLOCATE/SHARE GAMMA::$2$DIA3 DATA_DISK3

BETA$ DEVICESHARE ALLOCATE/SHARE EPSILON::$1$DUA1 DATA_DISK2
BETA$ DEVICESHARE ALLOCATE/SHARE GAMMA::$2$DIA3 DATA_DISK3

EPSILON$ DEVICESHARE ALLOCATE/SHARE BETA::DKA200 DATA_DISK1
EPSILON$ DEVICESHARE ALLOCATE/SHARE GAMMA::$2$DIA3 DATA_DISK3

GAMMA$ DEVICESHARE ALLOCATE/SHARE BETA::DKA200 DATA_DISK1
GAMMA$ DEVICESHARE ALLOCATE/SHARE EPSILON::$1$DUA1 DATA_DISK2
```

Figure 3–3 Shared Data Access



Data may be shared among users on multiple nodes, regardl where it physically exists on the network

Index