DeviceShare
Guide to Operations

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

Revision/Update Information: This is a revised manual.

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

Software Version: Version 2.0

Contents

Contents

Contents

Preface

The DeviceShare™ utility, engineered by Advanced Systems Concepts, Inc., allows all nodes in a DECnet network to share both tape and disk devices. DeviceShare is a significantly improved replacement for TapeShare. TapeShare™ V1.0 supported clusterwide tape sharing.

DOCUMENT PURPOSE

This operations manual describes the concepts and facilities of the DeviceShare utility and explains in detail the procedures for using this system.

INTENDED AUDIENCE

The intended audience for this manual consists of the following:

- Persons responsible for the installation of VMS-system-software layered products

- Persons responsible for the configuration management of VMS systems

---

™ DeviceShare and TapeShare are trademarks of Advanced Systems Concepts, Incorporated.

DOCUMENT STRUCTURE

The *DeviceShare Guide to Operations* consists of four chapters, an appendix, a glossary, and an index, as follows:

Section	Function
Chapter 1 OVERVIEW	Presents a general overview of the DeviceShare utility, its terms and concepts, and its features
Chapter 2 INSTALLING DEVICESHARE	Takes you through the installation in a step-by-step fashion, includes examples from an actual installation, and lists error conditions that might occur
Chapter 3 USING DEVICESHARE ON THE SERVER	Describes the commands you use to manage and monitor your served devices under the DeviceShare utility
Chapter 4 USING DEVICESHARE ON THE CLIENT	Describes the commands you issue to allocate and use the remote served devices under the DeviceShare utility
Appendix A MESSAGES	Lists all the user-interface messages for the DeviceShare system
GLOSSARY	Defines DeviceShare terms

CONVENTIONS

This document uses the following conventions:

- Square brackets ([ ]) indicate the enclosed item is optional.

- Uppercase indicates that you type text exactly as shown.

- Text in lowercase consists of data that you must supply—a device name, for example.

- All prompts and messages that the DeviceShare utility displays on your screen appear in small typewriter font, while all command syntaxes that you must enter appear in the same font, boldfaced, as follows:

  ```
  This is a message or a prompt.
  $ This is what you must enter.
  ```

- A comma, followed by a horizontal ellipsis (,...), indicates that you can supply more than one item, each separated by a comma.

- A vertical bar (|) indicates *one*, logical choice within a list, such as this *or* that.

- A [] indicates a key on your keyboard, for example, Return .

- A vertical ellipsis indicates the omission of system-supplied information, as in the following example:

  ```
  %VMSINSTAL-W-NOTSYSTEM, You are not logged in to the SYSTEM account.
  %VMSINSTAL-W-DECNET, Your DECnet network is up and running.
  %VMSINSTAL-W-ACTIVE, The following processes are still active:
      .
      .
      .
  * Do you want to continue anyway? [NO]   YES      Return
  ```

ASSOCIATED DOCUMENTS

For additional information consult the following documents:

- **Advanced Systems Concepts**

  Getting Started with DeviceShare

- **Digital Equipment Corporation**

  DCL Dictionary

  VAX/VMS System Messages and Recovery Procedures Reference Manual

  VMS BACKUP Utility Manual

  VMS I/O User's Reference Manual

  VMS License Management Utility Manual

  VMS VAXcluster Manual

If you are evaluating DeviceShare may we suggest reading the *Getting Started with DeviceShare* manual. This manual provides a very straightforward approach to evaluating the product and applying its many features and benefits to your systems.

1 OVERVIEW

1.1 WHAT IS DEVICESHARE?

DeviceShare is a system which allows all or specific nodes within a DECnet network to share selected tape and disk devices. You use DeviceShare to make a local tape or disk device accessible to other nodes in the network. By providing such access, DeviceShare expands the usefulness, connectivity, and availability of all your tape and disk drives. Once you designate a device for DeviceShare service, remote access can be easily established and the commands you issue to the remotely served device are the same as if the device was local.

The DeviceShare utility provides two methods for allowing systems to share devices; *DECnet* and *SCS*.

DECnet refers to Digital's Network Architecture and is the principal method that VAX nodes use to communicate with each other. When DeviceShare uses DECnet as its communications services, both tape and disk devices may be served to any other node in the network running DeviceShare. ASCI highly recommends that DECnet be used as the transport layer for maximum device accessibility.

SCS (System Communications Services) refers to Digital's VAXcluster communications services which allow nodes and devices to communicate with each other. Using SCS, DeviceShare uses the TMSCP protocol and establishes a connection to the other VAXcluster members. When using SCS, Digital's TUDRIVER is used as the standard tape device driver. Please read the release notes prior to using DeviceShare's SCS method, if the version of VMS you use is V5.5 or later. DeviceShare's use of SCS is limited to only tape drives.

1.2 TERMS AND CONCEPTS

1.2.1 Server and Client

Figure 1–1   Server and Client Nodes
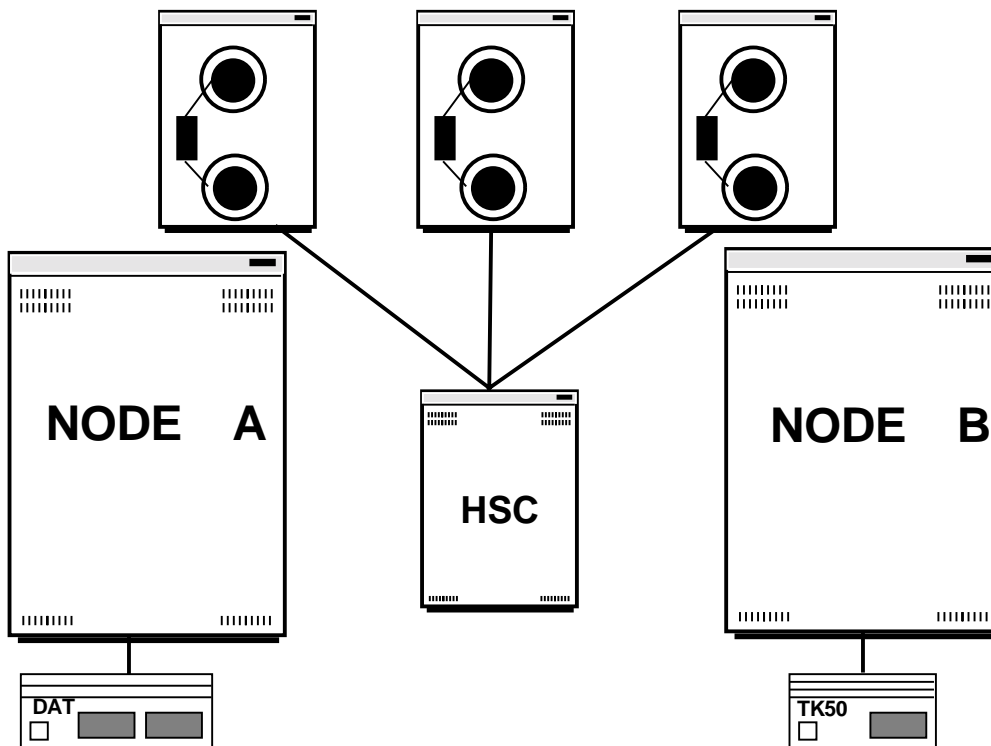
DeviceShare recognizes two kinds of nodes within a network—Server and Client.

A *Server* node is the system to which the device is physically connected. When the device is connected to an HSC controller, the Server system must be connected to the HSC as part of a Computer-Interconnect (CI) cluster. Figure 1–1 illustrates a Server system for a local DAT device (Node A), a local TK50 (Node B) and several HSC Tape drives (Nodes A and B).

A *Client* node is the system, *not* physically connected to the device. Figure 1–1 illustrates that Node B is a Client system in that it uses Node A's DAT drive. Likewise, Node A is a Client system in that it uses Node B's TK50. Client nodes use DeviceShare to transparently access served devices.

DeviceShare is installed and licensed for both Server and Client nodes. The Server license (product name DEVICESHARE) allows complete use of all DeviceShare facilities. The Client only license (product name DEVICESHARE-CLIENT) allows only user operations (ALLOCATE/DEALLOCATE) to be permitted.

1.3   RUNNING DEVICESHARE IN A NETWORK

Figure 1–2 depicts a network using DeviceShare services. Node A has a TA81 tape drive, Node B has a TA90 and several disk drives, Node C has a DAT drive and a local disk drive and Node D is a workstation with its own disk drives.

Figure 1–2   DeviceShare within a Network
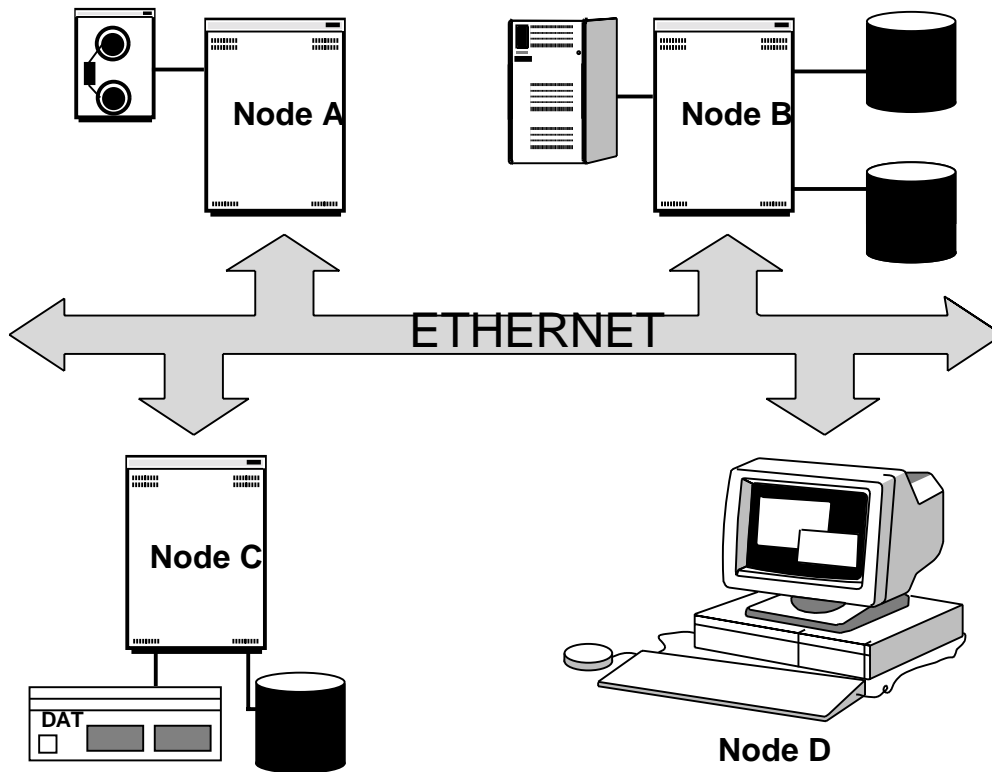
If all the nodes have Server licenses, then all tape and disk devices can be served to all the nodes in the network. If you wanted to only serve the TA81 tape drive, then Node A would have a Server license and Nodes B, C and D would have Client licenses.

1.3.1   Applications

DeviceShare solves many problems concerning the accessibility of remote tape and disk devices. The following list suggests some applications where DeviceShare can provide the greatest benefit. All examples reference Figure 1–2.

- **De-centralized Backup**.

  Using DeviceShare Nodes A, C and D can gain access to Node B's TA90 drive for the purposes of backup. The TA90 will appear to each node as a local tape device, and BACKUP will execute on each node, respectively, thereby utilizing the TA90 to its greatest potential while balancing the burden of backup among the various CPUs. Node B would have a Server license, and the other nodes would need Client licenses.

  If you wanted to allow all nodes to gain access to Node A's tape drive you would convert Node A into a server node.

- **Centralized Backup**.

  This approach is useful when backup procedures are already in place, as is usually the case for larger systems, and you simply want to backup additional disks. Node A, C, and D's disk drives would be served to Node B (since it has the TA90). That would allow Node B to perform all the disk backups to its TA90 and relieve the other nodes from backup considerations. Nodes A,C and D would have server licenses and Node B would have a client license.

- **Software Installation**.

  Since DeviceShare can serve both tape and/or disk, you can serve a CDROM reader to all nodes in the network (similiar to Digital's InfoServer, but less costly since you use existing equipment) and perform your installations from CD. This can result in less software maintenance since you don't need to get various distribution media. The other extreme to this situation, is receiving a software kit on TK50 that you need to install on a larger system. Rather than having to find space on your workstation to copy the savesets from TK50, and then find space on the target system for the actual installation, DeviceShare lets you serve the TK50 directly to the target system. Just execute VMSINSTAL and specify the TK50 drive directly. Simple and efficient!

- **Shared Information Access**.

  Quite often you may need to access a file(s) on a remote system, for either read or write purposes. While you can log into the remote system using SET HOST, you may need the processor power of the system you're on to complete your work. In addition, you may need to allow end-users to gain access to data and it would be preferable to keep things simple and avoid remote logins or DECnet node

specifications. For example, let's say you're a software engineering company and your products reside on several systems. One approach would be to require the user building a software kit to know the location (node) of each software product that could be built. The user would then need to log into the appropriate system to build the kit. Another approach might be to serve each node's product disk to all the node's in the network. That way you could create a *search list* containing all the product locations. So no matter what system you were on (both clustered and non-clustered systems), the same procedure would allow a user to build the desired kits.

Joint development by multiple workstation users (who are not all residing in the same VAXcluster) is another application for DeviceShare. A project's disks could be shared to all members of the group with full read/write access. Network access to those shared disks would be completely transparent to both users and programs. The disks would appear local to each system.

1.3.2   Security

DeviceShare provides the VMS System Manager with the ability to select which nodes may have access to a served device. DeviceShare provides two levels of security; Node and User-based.

Node security allows the System Manager to either specifically designate which nodes gain access, which nodes cannot gain access, or a combination of the two. For example, if Nodes A, B and C are in the network. You can restrict Node A's tape drive to only Node B through the following command:

```
$ DEVICESHARE SERVE/INCLUDE=B $1$MUA0:
```

Please note that complete wildcards are allowed as part of the INCLUDE and EXCLUDE qualifiers.

User-based security is provided to served disks which are shared to other nodes. You can select two (2) types of shared disks: Private and System-wide.

- **Private**

  The shared disk is mounted privately for this process, but allows other users on other nodes to share the disk volume. This feature requires that the client user possess a valid VMS username on the server system. All privileges, UIC, identifiers and disk quotas are taken from the user's VMS profile on the server system.

- **System-Wide**

  The shared disk is mounted system-wide and is available for use by all authorized users. This feature allows the server's system manager to set the security level required for access to the shared served disk.

  1 *Authentication.*

     The first level of security, this requires each client user to have a VMS account on the server system. The user is prompted for a password which together with the username are verified against the server system prior to gaining access to the disk.

  2 *Server.*

     The second level of security, this requires that the remote user's UIC and identifiers match those of the server system. DeviceShare will use that information whenever the remote user attempts to gain access to a file. All privileges, UIC, identifiers and disk quotas are taken from the server system.

  3 *Client.*

     The third level of security, DeviceShare will apply your remote UIC and privileges from the client system, when accessing the shared disk.

1.4   RUNNING DEVICESHARE IN A VAXCLUSTER

Figure 1–3 depicts a mixed-interconnect VAXcluster. In this configuration the magnetic-tape drives on the HSC are accessible only to Nodes A and B. Only Node B can access the DAT drive, and Node D alone can access its local disk drive. Node D has no tape drives at all.

Figure 1–3   A Mixed CI/NI VAXcluster
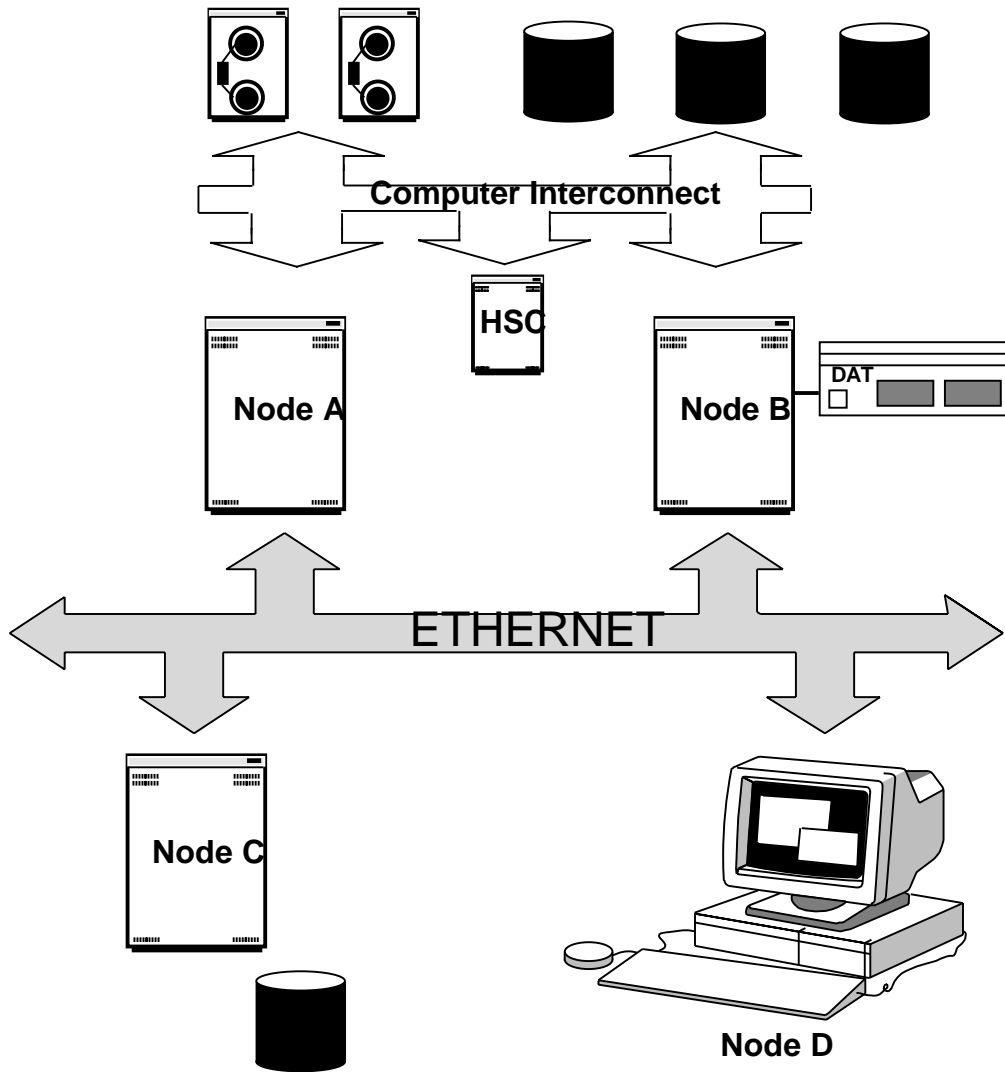
Installing DeviceShare on Node B permits every node in the cluster to access the magtape and DAT drives. Likewise, installing DeviceShare on node C allows full-cluster access to its local disk. Loading DeviceShare on Node A also provides cluster access to the HSC magtapes in the event you lose Node B.

Assume that the operators of Node D want to back up a locally attached disk to tape. As the cluster is currently configured in Figure 1–3, there is no direct means of accomplishing this.

When you install and run DeviceShare on the cluster, Node D becomes a DeviceShare Client. To make the DAT drive, on Node B, accessible to all cluster members running DeviceShare, issue the following command on Node B:

```
$ DEVICESHARE SERVE/SCS NODEB$MKA200:
```

Node B has become a DeviceShare Server.

1.5 DEVICESHARE FEATURES

DeviceShare provides great flexibility in terms of sharing tape and/or disk devices across the network. The following list provides an overview of the product's features.

- Network Access - DeviceShare serves tape and/or disk devices across the network. This allows much greater resource utilization of equipment than was previously possible. Both centralized and de-centralized backup strategies are supported, where tapes can be served to *tapeless* systems and disks can be served to large systems which already have integrated backup procedures.

- SCS Access - DeviceShare supports the use of Digital's SCS and TMSCP protocols for completely transparent serving of tape drives across a VAXcluster.

- Disk Serving - DeviceShare supports both exclusive and shared access to a disk drive with complete read/write capability. In exclusive mode, the disk is remotely served to your system, and no other system may access the drive. The disk can be manipulated in any manner, by all VMS utilities, as though it were a local disk drive. In shared mode, the disk must be mounted as a Files-11 volume on the serving system. You may read and/or write to all files, however, you cannot open a file for write-sharing.

- Automatic Device Serving - DeviceShare's startup procedure provides for both automatic and pre-set device serving. The automatic serving features allow for TAPE, DISK or BOTH. The pre-set device serving feature executes a DCL command procedure containing specific device SERVE commands which are issued at product startup.

- Generic Naming - DeviceShare supports the use of an ALIAS name (up to a maximum of three) which represents the served device. For example, TAPE or FAST_TAPE could represent a tape drive. The ALIAS feature allows your users to remember simple mnemonics which represent the device rather than the actual VMS device name.

- Generic Allocation - When multiple served devices use the same ALIAS, DeviceShare provides a facility where the first free device can be selected instead of selecting each drive in order to find the one that's available. For example, two tape drives named TAPE could be accessed directly, through a different alias or VMS devicename or allocated generically through the alias TAPE.

- Served Device Security - DeviceShare provides the Server System Manager the ability to specifically include or exclude one or more nodes from accessing a served device.

- Shared Disk Security - DeviceShare provides two (2) types of shared disk: Private and System-wide; and three (3) types of user level security: Authenticate, Server and Client.

- Network Data Compression - DeviceShare provides a data compression facility where repetitive data can be compressed during network transmission. This can significantly improve your effective throughput utilization, and is user specifiable. Data is compressed during network transfers only and is not compressed at the target device.

- Automatic Network Reconnection - DeviceShare will automatically restart on a network link failure and attempt to reconnect to the served system for fifteen (15) minutes (the default). If the reconnection succeeds then device I/O operations continue. This feature can prevent unnecessary repetition of work due to network failures.

- Remote Operator Communications - DeviceShare users can issue remote OPCOM commands similiar to VMS REPLY which can either send a message to an operator or request an action and reply. The operator's reply can be placed into a DCL symbol for program or command procedure analysis.

2 INSTALLING DEVICESHARE

2.1 PREPARING FOR INSTALLATION

The DeviceShare utility is a layered product for VMS Version 5.2 or later.

2.1.1 Release Notes

The DeviceShare utility provides online release notes that you can display or print.

They are available from the VMSINSTAL procedure, if you use the N option. VMSINSTAL can print and/or display the release notes.

After installation the release notes are in the file, SYS$HELP:DEVSHR020.RELEASE_NOTES, as Example 2–1 illustrates. In addition, online help provides the location of the release-notes file.

Example 2–1   Using Help to Find the DeviceShare Release Notes

```
$ HELP DEVICESHARE RELEASE_NOTES

DeviceShare

  Release_notes

     The release notes for DeviceShare V2.0 have been placed in the file
     SYS$HELP:DEVSHR020.RELEASE_NOTES.
```

2.1.2   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™*
- *DeviceShare™ Guide to Operations*
- 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.

2.1.3   Installation Checklist

Check each item in the following list before you start the installation:

1   A valid version of the VMS operating system (see Section 2.1.4)

2   A backup of your system disk (see Section 2.1.5)

3   An ASCI Product Authorization Key (PAK) for the type of DeviceShare license you have purchased (see Section 2.1.6)

4   Required privileges, quotas, disk space, and time (see Section 2.1.7)

5   A working familiarity with the VMSINSTAL procedure (see Section 2.1.8)

2.1.4   Prerequisite Software

The DeviceShare utility requires that you install a valid, VMS Version 5.2 (or later) operating system on your system before installing DeviceShare. To determine what version of VMS you have, type the following command:

```
$ SHOW SYSTEM

VAX/VMS V5.2  on node DeviceShare  10-OCT-1990 11:33:54.06   Uptime   6 22:25:03
  Pid     Process Name     State  Pri     I/O        CPU       Page flts Ph.Mem
```

The first line of the displayed information shows the VMS Version.

VMSINSTAL checks for a valid version and aborts the procedure, if it does not find one.

2.1.5   System-Disk Backup

ASCI recommends that you take the precaution of backing up your system disk before any installation. For instructions on backing up your system disk see the *VAX/VMS System Manager's Reference Manual*

2.1.6   Product Authorization Key Requirement

DeviceShare uses Digital's License Management Facility (LMF) as the
run-time portion of determining software license compliance. DeviceShare
is licensed in two forms: Server (full-function) and Client-only. A
DeviceShare Server license will allow all features of the product to be
used. A DeviceShare Client license will allow a remote node to use a
served device. However, that node cannot serve a device to other nodes.

When you received the product from ASCI or your local distributor it
should be accompanied by an ASCI Product Authorization Key (PAK).
This PAK is very similiar to the PAKs provided by Digital, and must
be registered and loaded prior to product use. The command procedure
@SYS$UPDATE:VMSLICENSE is normally used for PAK registration. ASCI
recommends that you register the DeviceShare PAK prior to the product
installation.

To determine whether you have the correct product license, please examine
the PAK you received. A product name of DEVICESHARE means you have
a Server license. A product name of DEVICESHARE-CLIENT means that
you have a Client-only license. If you are evaluating DeviceShare the
product name on the PAK may say "EVALUATION" and is equivalent to
a Server license. Evaluation PAKs, however, are valid for a limited time,
and the expiration date is printed in the "Product Termination Date" field
on the PAK.

Example 2–2 illustrates the use of the VMSLICENSE command
procedure.

Example 2–2   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.
```

Example 2–2 Cont'd on next page

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

```
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 ]

     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...
```

> Please read the *VMS License Management Utility Manual* for more
> information concerning LMF and VMSLICENSE.

2.1.7   VMSINSTAL Requirements

ASCI recommends that you run the VMSINSTAL procedure from the SYSTEM account.

The account you use for the installation must have the following privileges:

• CMKRNL

• TMPMBX

The account you use for the installation must have the following quotas:

• ASTLM = 24

• BIOLM = 18

• BYTLM = 18000

• DIOLM = 18

• ENQLM = 30

• FILLM = 20

The DeviceShare utility requires a peak utilization of 2250 blocks, and allocates a net amount of 775 blocks on your target device.

The installation takes about five (5) minutes from a magnetic tape, operating on a VAX 8250.

2.1.8   The VMSINSTAL Procedure

VMSINSTAL is a Digital-supplied, command procedure, used for installing software products from their distribution media to your system. To invoke the procedure, issue the following command:

```
$ @SYS$UPDATE:VMSINSTAL product device OPTIONS options_flag
```

You can invoke the command procedure, VMSINSTAL, without using any of the shown parameters (Table 2–1). If so, the procedure prompts you for the necessary information.

Table 2–1   VMSINSTAL Parameters

Parameter	Description
product	The product name and version—in this case, DEVSHR020
	This parameter is optional.
device	The device on which you mount the distribution volume
	The format is ddcu, dd is the device code, c is the controller, and u is the unit number. You can use a logical name that translates to a valid-device specification. This parameter is optional.
OPTIONS	A keyword parameter that indicates, whether you have used the options_flag parameter to specify any options
	This parameter is optional.
options_flag	A letter that specifies an option
	The only valid option for this installation is N, which allows you to print and/or view the release notes before the installation continues. This parameter is optional and must follow the keyword, OPTIONS.

The DeviceShare-installation process uses the standard VMS-installation procedure, VMSINSTAL. The DeviceShare product is normally distributed on 9-track, 1600-bpi magtape in BACKUP format.

The next section documents the installation procedure, step by step. Future versions of VMSINSTAL might contain new prompts that are not documented here; however, the installation should proceed normally.

2.2 PERFORMING THE INSTALLATION

Step 1: Log In

Log into the System Manager's account (SYSTEM) and use the operator's console terminal. Make sure that you do have all of the items, listed in the checklist in Section 2.1.3.

```
Username: SYSTEM      [ Return ]
Password:             [ Return ]
```

Step 2: Run VMSINSTAL

Invoke the installation-command procedure as follows:

```
$ @SYS$UPDATE:VMSINSTAL DEVSHR020 $2$MUAO: OPTIONS N     [ Return ]
```

DEVSHR020 is the product name and version.

2MUA0: represents a valid device name on which you mount the distribution media. If the media is a disk, use the directory and device names.

OPTIONS N indicates that you want VMSINSTAL to prompt you for the release-notes options.

If you do not specify a product or device, the system prompts you for them.

If you do not want to type or print the release notes before the installation, omit OPTIONS **N**. After the installation you can find the release notes in the SYS$HELP directory.

To abort the installation, type [CTRL] [Y]. The installation procedure then deletes any files you have created up to that point and exits. To restart the installation, proceed with Step 2:.

Step 3: Check Warnings

VMSINSTAL might display any or all of the following warning messages and ask if you want to continue.

```
%VMSINSTAL-W-NOTSYSTEM, You are not logged in to the SYSTEM account.
%VMSINSTAL-W-DECNET, Your DECnet network is up and running.
%VMSINSTAL-W-ACTIVE, The following processes are still active:
   .
   .
   .
* Do you want to continue anyway? [NO]  YES    Return
```

If you choose not to continue, you return to the DCL prompt. Correct the warning condition, and restart with Step 2: above.

Step 4: Create a Backup

If you have not backed up your system disk, do not continue with the installation. If you need to back up the system disk, type **NO**. VMSINSTAL exits and returns you to the DCL prompt. Backup your system disk and start VMSINSTAL at Step 2: above.

```
* Are you satisfied with the backup of your system disk? [YES]
YES    Return
```

Step 5: Load the Distribution Media

VMSINSTAL asks you to load the distribution media onto the device.

```
Please mount the first volume of the set on $2$MUA0:.
* Are you ready? [YES]  YES    Return
%MOUNT-I-MOUNTED, DEVSHR020 mounted on _$2$MUA0: (HSC000)
```

Step 6: Select the Release Notes Option

This step applies only if you specified the OPTIONS **N** in Step 2:.

```
Release Notes Options:

 1.   Display release notes
 2.   Print release notes
 3.   Both 1 and 2
 4.   Copy release notes to SYS$HELP
 5.   Do not display, print or copy release notes

* Select option [2]:   2     Return
* Queue name [SYS$PRINT]:   SYS$PRINT     Return
Job DEVSHR020 (queue SYS$PRINT, entry 1117) started on SYS$PRINT


* Do you want to continue the installation? [YES]
   YES     Return
```

Option 1 immediately displays the release notes at your terminal. Option 2 prompts you for a print queue and spools the release notes for printing. Option 3 does both. Option 4 copies the release notes to the SYS$HELP directory, and Option 5 prevents VMSINSTAL from displaying, printing, or copying them.

Step 7: Purge Existing Files

The DeviceShare utility creates a new generation of these system files during installation:

- SYS$LIBRARY:DCLTABLES.EXE

- SYS$STARTUP:DEVICESHARE_STARTUP.COM

- SYS$STARTUP:DEVICESHARE_STARTUP_LOCATION.COM

VMSINSTAL displays the following prompt, asking whether you want to purge old versions of these files:

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

Step 8: Choose a Product Location

To store most of the DeviceShare product files, you are prompted for a
device and directory, which is created for you, if one does not already exist.
The default location is SYS$SYSDEVICE:[DEVICESHARE].

```
DeviceShare requires a location for its files.
* Please enter a device/directory? SYS$SYSDEVICE:[DEVICESHARE]    Return
```

Note: **When all Server and/or Client nodes use a common-system disk,
you need to install DeviceShare only once.**

Step 9: Choose the Installation Type

At this point you need to distinguish, where you want to install
DeviceShare—whether on a Server or on a Client node. Remember, a
Server node includes client functionality. VMSINSTAL asks the following
question:

```
* Is this a CLIENT-only installation?  [NO]    NO      RETURN
```

Answer *NO* (the default), if you're using the software for both Server
and/or Client systems. Answer *YES*, if a Client-only system is licensed for
use.

Step 10: Enter License Product Authorization Key

At this point you need to confirm that you have loaded the proper DeviceShare Product Authorization Key. As discussed in Section 2.1.6, DeviceShare is licensed in two forms: Server (full-function) and Client-only.

The installation process will display the name of the license that it expects you to have based on your answer to the previous question. A Server license is named DEVICESHARE, a Client-only license is named DEVICESHARE-CLIENT.

A Server license installation will appear as:

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

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

A Client-only license installation will appear as:

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

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

If you have loaded the appropriate license onto your system, please enter YES. If you didn't register the license yet, but you physically possess the ASCI PAK, you may still enter YES. However, you will not be able to use the product until you have registered the PAK. If you don't possess the product PAK, you should abort this installation by entering NO, and obtain a PAK from ASCI or your local distributor.

Note: Steps 11 and 12 are performed for a Server installation Only.

Step 11: Create Server Account

The DeviceShare utility creates a special account for its DECnet usage. The account is heavily restricted and can be used only by the DeviceShare server program, DEVSHR_NTS. The next query requests a password for the account. The password is re-verified since it will not echo.

```
* Enter the PASSWORD for DEVSHR_NTS account (minimum 12 characters):
* Verify:
```

Next, the procedure requests a unique UIC as part of the account creation.

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

Finally the procedure requests a device for the directory DEVSHR_NTS, where special audit logs are kept concerning device usage.

```
* Enter the DEVICE where the DEVSHR_NTS directory will reside [SYS$SYSDEVICE]:
```

Step 12: Define DECnet Network Object

DeviceShare now defines the Server network object in both the permanent and volatile databases.

Step 13: Allow the Installation to Complete

You are not prompted again until the installation completes successfully. Then it displays the following messages:

```
        Installation of DEVSHR V2.0 completed at 12:20

 VMSINSTAL procedure done at 12:20
```

Step 14: Log out

Note that VMSINSTAL deletes or changes entries in the process-symbol tables during the installation. Therefore, if you continue to use the System Manager's account and want to restore those symbols, log out and log in again.

```
$ LOGOUT    Return
SYSTEM      logged out at 10-SEP-1990  12:21:00.00
```

The installation procedure adds a new command, DEVICESHARE, to your system's DCLTABLES. The installing process will have access to the DeviceShare command as soon as the installation successfully completes. All other processes will need to logout and back in to acquire the new command.

Please note that rebooting a system also makes the DeviceShare command available for use.

2.2.1   Error Conditions

If the installation procedure fails for any reason, VMSINSTAL displays the following message:

```
%VMSINSTAL-F-UNEXPECTED, Installation terminated due to unexpected event.
```

This unexpected event can result from any of the following conditions:

- Insufficient disk space to complete the installation

- Insufficient AST quota

- Insufficient buffered-I/O-byte count

- Insufficient subprocess quota

- Insufficient open-file quota

- Insufficient process-paging-file quota

- Insufficient process-working-set quota

- Insufficient system-maximum-working set

- Incorrect version of VMS

For descriptions of the error messages, generated by these conditions, see the *VAX/VMS System Messages and Recovery Procedures Reference Manual* and the *Guide to VAX/VMS Software Installation*.

If you are notified that any of these conditions exist, take the appropriate action, as described in the message. You might need to change a system parameter (with SYSGEN) or increase an authorized-quota value (with AUTHORIZE). If the installation fails, restart the installation procedure from Step 2:.

2.2.2   Sample Installation

The following sample installation illustrates the sequence of the various
VMSINSTAL prompts and queries and their sample answers. The sample
installation is for a Server node although the Client installation differs
very slightly.

Example 2–3   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]? YES   Return

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 2–3 Cont'd on next page

Example 2–3 (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
```

2.2.3   Installation Messages

The messages that follow are DeviceShare product-specific messages, which can be displayed while you're installing the product. Any message not listed in this section should be investigated by examining the manual *VAX/VMS System Messages and Recovery Procedures Manual.*

BADVMSVER,   This kit requires VMS V5.0 or later

Facility: VMSINSTAL

Severity: Fatal

Explanation: You must install DeviceShare on a VMS V5.0 or later system.

User Action: Self-explanatory

DEVNOTAVL,   Device *device-name* is not available

Facility: VMSINSTAL

Severity: Error

Explanation: The device you entered is not currently available.

User Action: Make the device available or choose another available device and rerun the installation procedure.

DEVNOTDISK,   Device *device-name* is not a disk

Facility: VMSINSTAL

Severity: Error

Explanation: The device you entered is not a disk drive.

User Action: Select a disk drive and rerun the installation procedure.

DEVNOTEXIT,   Device *device-name* does not exist

Facility: VMSINSTAL

Severity: Error

Explanation: The device you specified is unknown to the system.

User Action: Specify a legal-disk device and rerun the installation procedure.

DEVNOTMNT,   device *device-name* is not mounted.

> **Facility:** VMSINSTAL
>
> **Severity:** Error
>
> **Explanation:** The device you entered is not mounted or is not mounted Files-11.
>
> **User Action:** Mount the disk as a Files-11 volume and rerun the installation procedure.

NOSPACE,   device *device-name* contains only *nnn* free blocks. At least *nnn* blocks are required.

> **Facility:** VMSINSTAL
>
> **Severity:** Error
>
> **Explanation:** The device you entered has insufficient free-disk space upon which to install the product.
>
> **User Action:** Delete noncritical files to make room, or choose another disk device, and rerun the installation procedure.

2.3   VAXcluster Post Installation

After you have installed DeviceShare on a member of the cluster, you may
need to perform one or more steps to complete the installation for other
cluster members. The following questions indicate which sections you may
need to execute for each additional cluster member.

**1   Does your license allow you to execute multiple copies of
DeviceShare?**

No - STOP and purchase additional PAKs from ASCI or your local
distributor.

2   Do you have a common system disk?

No - read and execute procedure described in Section 2.3.1.

3   Does the file SYS$COMMON:[SYSEXE]NETOBJECT.DAT **exist?**

No - read and execute the procedure described in Section 2.3.2.

Please remember that if you installed DeviceShare on a non-system disk,
you must have that disk mounted and available to the cluster, to be able
to run DeviceShare on multiple nodes of a VAXcluster.

2.3.1   Create DeviceShare Account

The following procedure must be executed for a cluster member *if* that
node is to be a **Server** system.

```
$ @DEVICESHARE_PRODUCT:DEVSHR_NTS_DCLOBJ   Return

    1.   Create DEVSHR_NTS account

    2.   Declare DEVSHR_NTS object name

  999.  Exit

Please enter desired option: 1   Return
+----------------------------------------------------------------------+
!                                                                      !
! 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$SYSDEVICE]
%UAF-I-ADDMSG, user record successfully added
%UAF-I-RDBADDMSGU, identifier DEVSHR_NTS value: [003342,003342] added to rights dat
%UAF-I-RDBADDMSGU, identifier DEVSHR value: [003342,177777] added to rights data ba
%UAF-I-MDFYMSG, user record(s) updated
%DEVICESHARE-I-CREDIR, Creating SYS$SYSDEVICE:[DEVSHR_NTS] directory

%DEVICESHARE-I-MODUAF, Modifying account
%UAF-I-MDFYMSG, user record(s) updated
%UAF-I-MDFYMSG, user record(s) updated
%UAF-I-MDFYMSG, user record(s) updated
%UAF-I-MDFYMSG, user record(s) updated
```

This procedure will create a special VMS account for DeviceShare use.
The account is enabled for DECnet use only and is heavily restricted.
If the account already exists, this procedure will ensure that it is setup
properly.

2.3.2   Define DeviceShare DECnet Object

The following procedure must be executed for a cluster member *if* that
node is to be a **Server** system.

```
$ @DEVICESHARE_PRODUCT:DEVSHR_NTS_DCLOBJ   Return

    1.   Create DEVSHR_NTS account

    2.   Declare DEVSHR_NTS object name

  999.   Exit

Please enter desired option: 2   Return
Enter PASSWORD for DEVSHR_NTS login:
```

This procedure will define a DeviceShare object named DEVSHR_NTS in
both the permanent and volatile databases. Please note that by default
proxy accessing is set to *incoming* for simpler usage of the product. You
may change the setting to *none* for additional security.

2.4   PRESTARTUP PROCEDURES

After you have installed DeviceShare you must determine the Tape
Allocation Class value that you want to set for each *Server* node, if you
plan to use the SCS method of serving tape drives. If you do not plan on
using this feature, please skip to Section 2.5.

2.4.1   Rules for Determining the Tape Allocation Class Value

Note: **This section should be read if you plan on using the SCS method
of serving tape devices.**

Each *Server* node must have a Tape Allocation Class value. The rules for
determining this value are, as follows:

- VAX's that are connected to an HSC through the CI to which you want
  to serve HSC-connected-tape devices must use the same nonzero, Tape
  Allocation Class value.

- All cluster-accessible tape devices on nodes with a nonzero Tape
  Allocation Class value must have unique device names.

- Local tape devices with a Tape Allocation Class value of zero can have
  the same unit number on different-cluster nodes.

Zero (0) is the default, Tape Allocation Class value. Assign this value to
any node in an NI (LAVC) cluster. In a mixed-interconnect or CI-only
cluster, however, all the following must have a nonzero Tape Allocation
Class value:

- HSC's

- Systems serving HSC tapes

Warning: **If you assign Tape Allocation Class values incorrectly, you might
create lock conflicts.**

2.4.1.1   Setting the Tape Allocation Class Value

Prior to issuing the DEVICESHARE_STARTUP command procedure, you must set the Server node's Tape Allocation Class SYSGEN parameter, TAPE_ALLOCLASS. This is a static parameter so a reboot of your system will be necessary for the parameter change to take effect. ASCI highly recommends using AUTOGEN for all SYSGEN parameter changes.

2.5 STARTING DEVICESHARE

Normally, you load the DeviceShare utility and have it ready
for DeviceShare commands, when you bring up your system. A
command procedure, named DEVICESHARE_STARTUP, found in the
SYS$STARTUP directory, initiates the system. To run successfully,
DEVICESHARE_STARTUP requires several privileges (Table 2–2).

Table 2–2   DEVICESHARE_STARTUP Privileges

CMKRNL	OPER	PSWAPM
SYSNAM	SYSPRV	TMPMBX
PRMMBX	WORLD	LOG_IO
PHY_IO	NETMBX	PFNMAP
EXQUOTA	CMEXEC	DETACH

If you execute this command procedure within your VMS startup
procedure, you should not have any problem with these privileges.

If the DeviceShare product resides on a disk device other than
SYS$SYSDEVICE, you must mount that disk system-wide, prior to
invoking this procedure.

DEVICESHARE_STARTUP performs several tasks:

1   Determines the location of the product

2   Performs various checks determining whether the product can be
loaded based on; VAXcluster requirement if SCS selected, non-page
pool requirements of 5000 free bytes (actual or potential)

3   Loads the proper DeviceShare-device driver(s), based upon whether it
is a Server or Client-only system, and whether you want SCS and/or
DECnet transport methods

4   Installs DeviceShare images with privileges necessary to perform their
tasks

5   Performs automatic device serving for Tape and/or Disk devices, if you
select that option

6   Executes a DCL procedure (SYS$STARTUP:DEVICESHARE_SERVE.COM)
which allows you to serve and/or unserve specific devices

2.5.1   DeviceShare Startup Options

To start DeviceShare, execute the following DCL command:

```
$ @SYS$STARTUP:DEVICESHARE_STARTUP P1 P2 P3 P4
```

For a Server startup, the *P1* parameter should be **SERVER**. For a Client-only startup, the *P1* parameter should be **CLIENT**. If *P1* is omitted, a SERVER startup is assumed.

The *P2* parameter indicates the transport methods you want DeviceShare to support. You may choose **SCS**, **NETWORK** or **BOTH**. If *P2* is omitted, the NETWORK method is loaded.

The *P3* parameter indicates whether you want automatic network device serving on DeviceShare startup. You may choose **TAPE**, **DISK**, or **BOTH**. If you choose TAPE, then all tape class devices will be automatically served using the NETWORK method depending on your specification of the *P1* parameter explained above. If you choose DISK, then all disk class devices will be automatically served using the NETWORK method (unless you indicated that only SCS is to be loaded). If you choose BOTH, then all tape and disk class devices will be served. If you omit the *P3* parameter, then no devices will be automatically served. Please note that you may also manually serve devices through the DEVICESHARE_SERVE.COM procedure. This DCL procedure is invoked after DeviceShare has been started to allow you to indicate which specific devices are to be served.

If *P3* is specified, then you may also indicate additional qualifiers to be appended to the SERVE command, as part of the automatic device serving, through your specification using the *P4* parameter. For example:

```
$ @SYS$STARTUP:DEVICESHARE_STARTUP SERVER NETWORK TAPE "/EXCLUDE=SCOTTY"
```

This example causes a Server DeviceShare startup using the Network transport method only. All Tape devices are to be automatically served, and node SCOTTY is to be excluded from using those devices.

The startup procedure installs most DeviceShare utility programs so that your system operators do not require extraordinary VMS privileges (i.e. CMKRNL). For more information read the comments in the DEVICESHARE_STARTUP command procedure itself.

ASCI recommends that you issue your DEVICESHARE SERVE commands (see Section 3.2.2) immediately following DeviceShare start up.

2.5.2   DeviceShare Parameters

DeviceShare provides a facility for customizing the network parameters used by the product. Since all customer networks are different, three (3) parameters have been provided for your use. The parameters described in the table below all begin with the prefix DEVICESHARE_TAPE_ (for example, DEVICESHARE_TAPE_CMMSG). These parameters are to be defined as system-wide, logical names. For example,

```
$ DEFINE/SYSTEM DEVICESHARE_TAPE_CMMSG 16536
```

This command would change the maximum network QIO size to 16536 bytes. If your system issues large tape I/Os, a large network QIO size results in less segmentation at the QIO level and less resulting CPU utilization. These DeviceShare parameters are only applicable for Client usage (both Server nodes which also need to act as Clients *and* Client-only nodes). These logical names must be defined *prior* to executing DEVICESHARE_STARTUP.

Table 2–3   DeviceShare Client Logical Name Parameters

Parameter	Default	Usage
CMMSG	8192 bytes	This parameter controls the maximum network QIO size. A range of 8192 to 65537 is provided. Larger network QIO sizes can result in less CPU utilization. An overly large network QIO size results in wasted virtual memory.
CXBUF	256k bytes	This parameter controls the size of the network transmit buffer pool. DeviceShare will attempt to send as many buffers to the Server system as it can asynchronously. The minimum value is 65k bytes. The CXBUF parameter should be completely divisible by CMMSG parameter. The default should serve most purposes.
CRTMR	900 seconds	This parameter controls the length of time DeviceShare will attempt reconnection to a server node before abandoning the restart attempt. DeviceShare attempts to reconnect to the Server node in the effect of a network link failure. If the Server system is rebooted, the reconnection will ultimately fail and the I/O operations will be posted in error. A value of zero (0) indicates that no network reconnection should be attempted if the network link fails.
CXTMR	0 milliseconds	This parameter controls the length of time DeviceShare will attempt to wait for multiple I/O requests and multiplex those requests into a single DECnet message. A time of zero (0), the default, means that each I/O request is sent as a separate DECnet message. A non-zero value means that DeviceShare will wait up to n milliseconds before shipping as many requests as were queued during that time. Setting this value too high can have an adverse impact on performance.

2.5.3 SYSGEN Parameters

You might need to adjust the SYSGEN parameters below for proper DeviceShare operation. While the list presented is not exhaustive, due in large part to the complex interrelationship between many parameters, it represents the major SYSGEN parameters that have direct application to DeviceShare.

Table 2–4 Major SYSGEN Parameters

Parameters	Use	Adjustment Instructions
CHANNELCNT	Maximum Channels per process	This parameter, may need to be increased if you are serving many users with open files to a shared disk. The current VMS default (V5.5) is 128 channels per process. The current maximum is 2048 channels.
NPAGEDYN	Various VMS and DeviceShare purposes	DeviceShare requires varying amounts of nonpaged pool. The best technique for adjusting this parameter is to use the system and observe the VMS extension of nonpaged pool, IRP, SRP, and LRP with the DCL SHOW MEMORY/POOL command. You can then use these values as approximate starting points.
NPAGEVIR	Various VMS and DeviceShare purposes	This should be set significantly higher than NPAGEDYN.
NUMSPTE	Various VMS and DeviceShare purposes	DeviceShare uses 128 SPTE's if you select the SCS method of communications. The normal VMS default value is usually sufficient.
TAPE_ALLOCLASS	Tape Allocation Class	This parameter, should be set to the appropriate value, as discussed in Section 2.4.1.1.

3 USING DEVICESHARE ON THE SERVER

The DeviceShare utility uses standard-DCL syntax for its commands and their parameters and qualifiers.

3.1 HELP TEXT

The DeviceShare utility-installation process adds the necessary help text to your system-help library. For online help, type the following command:

```
$ HELP DEVICESHARE

DeviceShare

   The DeviceShare system, produced by Advanced Systems Concepts, Inc.,
   allows you to share local tape and devices with other nodes in the
   network or VAXcluster.

   Format:

   DEVICESHARE function/qualifier parameters

   DeviceShare commands are divided into two categories.  Those commands
   which can be executed on the Server and those commands which can be
   executed on the Client.

  Additional information available:
   Client     Server     Release_notes
DEVICESHARE Subtopic? SERVER
DeviceShare

  Server

    DeviceShare is typically used on the Server system to indicate which
    devices may be served and to which nodes.  The commands which follow
    discuss all commands that may be specified on the Server node.

   Additional information available:

   MODIFY     MONITOR    SERVE      SHOW       UNSERVE
```

3.2   PERFORMING DEVICESHARE OPERATIONS

The DeviceShare utility allows you to perform these operations:

- Make a local disk/tape device accessible to the network
- Make a local-tape device accessible to the cluster
- Remove a local-disk/tape device from network accessibility
- Display tape devices that DeviceShare serves
- Display nodes that DeviceShare serves
- Monitor DeviceShare activities

Figure 3–1   Server Client Use
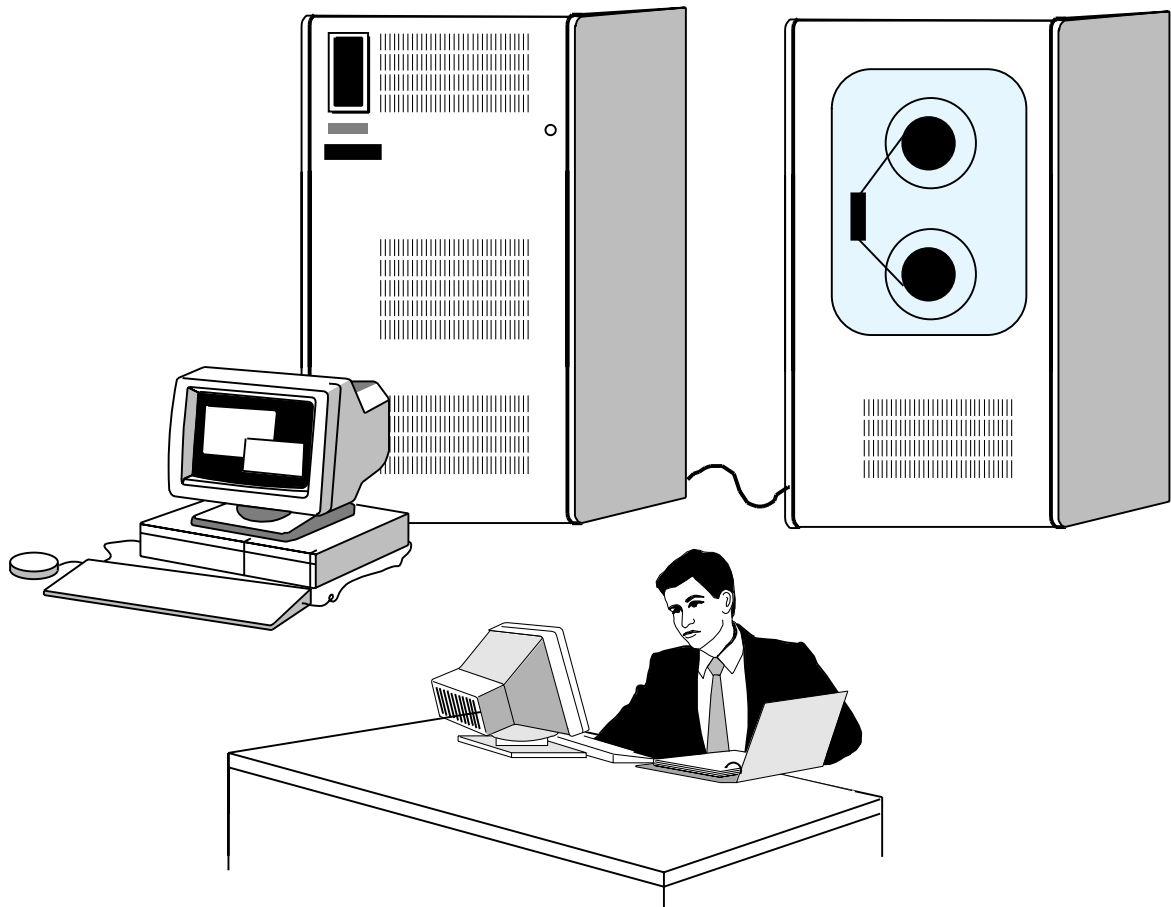
3.2.1   Making a Device Remotely Accessible

The DeviceShare SERVE/NETWORK command makes a local device
accessible to other nodes within a network. You must issue this command
from the device's Server system—that is, the system to which the device is
physically connected.

To make a tape or disk device remotely accessible, issue the following
command:

```
$ DEVICESHARE SERVE/NETWORK device_name
```

3.2.1.1   ALIAS Names and Generic Device Allocation

DeviceShare allows you to provide a *shorthand name*, called an ALIAS,
to reference a device. DeviceShare will always create an ALIAS using
the device type when you serve the device. You can specify up to three
(3) more ALIAS names of your own choosing. Typically, you will want to
select a short descriptive name that references the device you want to
serve. A good convention to follow might be to have an alias of TAPE for
every node that has at least one reel-to-reel type tape. That way, any user
in your network could simply reference the remote tape drive through its
alias of TAPE without knowing the actual physical device name.

DeviceShare also permits two or more devices to share an ALIAS name.
For example, FAST_TAPE could refer to two (2) TA81 tape drives.
1MUA0: and 1MUA1: are their respective device names. Without
the generic allocation feature, your remote user would first have to try
and allocate 1MUA0:, and assuming it's in use, issue a second allocation
request to 1MUA1:. With the generic allocation feature, DeviceShare
will provide the first available drive named FAST_TAPE to your user.

3.2.1.2   Network Communications Failure

DeviceShare provides an automatic restart feature for all served devices
so that if a transient network failure occurs, DeviceShare can continue
operations. This feature is particularly useful when you are in the middle
of a long backup and a network failure occurs. DeviceShare will, by
default, wait up to fifteen (15) minutes while attempting to reconnect. If
the timeout period expires, all access to the device will yield a *Device not in
Configuration* error. If the reconnection is successful then all operations to
the device continue. DeviceShare employs a sequence numbering scheme
to ensure that all I/O requests have been transmitted or received over the
network. Section 2.5.2 provides more information concerning DeviceShare
parameters.

3.2.1.3 Shared Disk Security

When you serve a disk device, you can elect the security requirements that must be gained prior to a remote user's successful I/O operation. DeviceShare defines a shared disk as one which is *already mounted* as a system-wide Files-11 device. The shared aspect is that multiple nodes and users can all access the device simultaneously. As the VMS System Manager you can select the security level that a remote user must gain before being able to successfully perform file operations. DeviceShare provides three (3) levels to choose from: Authenticate, Server and Client.

- *Authenticate*

  Authenticate is the highest form of security and requires the remote user to have a valid VMS username on the server system. DeviceShare will require the user to enter his username/password prior to accessing the shared disk. If the username/password are correct, all privileges,identifiers and disk quotas associated with that user will be used whenever file operations are performed. If the remote user does not authenticate himself, the remote access is rejected.

- *Server*

  Server is the second level of security and *assumes* that the username and UIC on the remote system are the same are those on the server system. If true, DeviceShare will use the server's username account profile (from SYSUAF.DAT) to ensure proper file security. The user's privileges, indentifiers and disk quotas from the serving system will be used whenever access is made to the shared disk.

- *Client*

  Client is the third level of security and propagates the user's remote UIC and privileges when accessing the shared disk.

It is important to note that shared disk security levels are only used when a disk is allocated by a remote system for system-wide usage. DeviceShare does allow each individual user to gain access to a shared disk in a node-specific private fashion. Access gained this way *requires* the user to also specify access control information (username/password or username, if proxy access is allowed) as part of the ALLOCATE command. In this case, all privleges, identifiers and disk quotas are those associated with the user's security profile on the server system (similiar to the SERVER security level).

3.2.2   Making a Tape Cluster Accessible

The DeviceShare SERVE/SCS command makes a local-tape device accessible to other nodes within a VAXcluster. You must issue this command from the device's Server system—that is, the system to which the device is physically connected.

To make a tape device accessible, issue the following command:

```
$ DEVICESHARE SERVE/SCS device_name
```

SERVE

This command makes a local tape or disk device accessible to other nodes in the network. A local tape can also be served to other nodes in a VAXcluster, through SCS services.

FORMAT

DEVICESHARE SERVE   device_name

Command Qualifiers	Defaults
/ALIAS=list	See Text
/[NO]COMPRESS	/NOCOMPRESS
/EXCLUDE=list	None excluded
/INCLUDE=list	/INCLUDE=*
/NETWORK	/NETWORK
/SECURITY=keyword	/SECURITY=SERVER
/SCS	/NETWORK

prompts

Device_name: device_name

restrictions

- This command requires PHY_IO, OPER, VOLPRO, and MOUNT privileges.

- FOR SCS USE: The device must be a tape-class device that is not already served and has the same allocation class as that set for this node. The NETWORK and SCS qualifiers are mutually exclusive.

- You must not currently mount, allocate, or have any open channels for tape devices.

PARAMETERS

device_name
Specifies the name of the device which you want served

DESCRIPTION

This command enables a device to be served to other client systems.

If the NETWORK qualifier is specified the device can be a disk or tape device and will be servable to other remote nodes in your network running the DeviceShare software. If NETWORK or SCS qualifiers are omitted, then NETWORK is used as the default.

If the SCS qualifier is specified then SCS services are used. The device must a tape-class device, which will be MSCP-servable to other nodes in the VAXcluster. You must issue this command on the Server node. In addition a tape device must not be currently mounted, allocated, or have any open channels.

COMMAND QUALIFIERS

/ALIAS=list

The ALIAS qualifier, valid for NETWORK served devices, allows you to specify up to three (3) pseudonyms, by which other remote nodes can reference the device. An alias can be a maximum of fifteen (15) characters. DeviceShare always creates an alias based on the device's type (i.e. TU81, TK50). These aliases can be used by other remote nodes as part of the ALLOCATE command. Alias names do not have to be unique and this flexibility allows a generic naming facility. For example, if you plan to serve two tape drives which can both operate at 6250bpi, you might want to provide an alias of 6250BPI for both devices. This would allow a remote client system to simply request any available 6250BPI device rather than a specific device.

/[NO]COMPRESS
/NOCOMPRESS (default)

This qualifier, valid for NETWORK served devices, allows you to set the default characteristics for network data compression. Network data compression reduces the amount of network traffic by compressing characters which occur at least 4 consecutive times into a three byte sequence. Depending on the content of your data this can significantly increase the throughput and overall speed of your tape operations. Please note that data compression is only provided across the network, data written to the served device is not compressed.

/EXCLUDE=list
/EXCLUDE=none (default)

This qualifier, valid for NETWORK served devices, allows you to exclude one or more nodes from using this device. Each node entry may use wildcard characters for more flexibility. To change this list at a later time, please use the MODIFY command. By default, no nodes are excluded from using this device.

/INCLUDE=list
/INCLUDE=* (default)

This qualifier, valid for NETWORK served devices, allows you to include one or more nodes that are allowed to use this device. Each node entry may use wildcard characters for more flexibility. To change this list at a later time, please use the MODIFY command. By default, all remote nodes may use the device.

/NETWORK
/NETWORK (default)

This qualifier indicates that the tape or disk device is to be served to other eligible nodes in the network. DECnet is used as the intersystem communications protocol. If the NETWORK or SCS qualifiers are omitted, NETWORK services are used.

The SCS and NETWORK qualifiers are mutually exclusive and cannot be specified together. Serving a device through both SCS and NETWORK services is supported, however, you must execute two SERVE commands to achieve the desired effect.

/SECURITY=keyword
/SECURITY=SERVER (default)

The SECURITY qualifier, observed for *shared* disks, allows you to enforce a level of security for remote users of your disks. Three (3) security levels are available: AUTHENTICATE, SERVER and CLIENT. AUTHENTICATE is the most restrictive form of security and requires the remote user to first authenticate himself prior to accessing the disk. SERVER expects the remote user to have a valid VMS authorization record and the same UIC as the local and remote systems. SERVER does not propagate the user's remote privileges when accessing files on the shared disk. CLIENT is very similiar to SERVER except the user's remote privileges *are* used when accessing files on the shared disk. See Section 3.2.1.3 for more information. By default, the security level for a system shared disk is SERVER.

/SCS
/NETWORK (default)

If the SCS qualifier is specified then SCS services are used. SCS is only supported in a VAXcluster environment. The major advantage to using the SCS protocols rather than DECnet is significantly improved performance.

The SCS and NETWORK qualifiers are mutually exclusive and cannot be specified together. Serving a device through both SCS and NETWORK services is supported, however, you must execute two SERVE commands to achieve the desired effect.

EXAMPLES

1 `$ DEVICESHARE SERVE $2$MUA0`

This example makes the tape device, 2MUA0, accessible to the other nodes in the network.

2 `$ DEVICESHARE SERVE/ALIAS=(FAST,6250BPI) MUA0`

This example makes the tape device, MUA0, accessible to the other nodes in the network. Three (3) aliases are available when referencing this device, TU81 (the default alias), FAST and 6250BPI.

3 `$ DEVICESHARE SERVE/EXCLUDE=SCOTTY $1$DUA10:`

This example makes the disk device, 1DUA10, accessible to the other nodes in the network except SCOTTY.

4 `$ DEVICESHARE SERVE/SCS $2$MUA0`

This example makes the tape device, 2MUA0, accessible to the other members of the cluster.

5 `$ DEVICESHARE SERVE/SECURITY=SERVER $1$DUA11:`

This example make the disk device, 1DUA11, accessible to other nodes in the network. SERVER level security will be enforced for any system-wide disk sharing on the remote nodes.

3.2.3   Removing a Device from Network Accessibility

The DeviceShare UNSERVE command removes an available device from
network accessibility. This command, issued on the server system, must
reference a device which is already served to the network via DeviceShare
and is not currently in use.

To remove a tape or disk device from network accessibility, issue the
following command:

```
$ DEVICESHARE UNSERVE device_name
```

UNSERVE

This command removes a local disk or tape device from network accessibility. Other nodes in the network will no longer be able to access the previously served device. SCS served devices cannot be unserved due to restrictions in VMS.

FORMAT　　**DEVICESHARE UNSERVE　[device_name]**

Command Qualifiers	Defaults
/ALL	

prompts　　Device_name:　　device_name

restrictions
- This command requires OPER and MOUNT privileges.
- The device must be currently served through DeviceShare and cannot be in use.

PARAMETERS　*device_name*

Specifies the name of the device which you want unserved. This parameter is required unless the ALL qualifier is specified.

DESCRIPTION

This command allows a system manager to remove a device from network accessibility. This command is particularly useful when your local system(s) need to access the local devices for various standalone or exclusive use operations. The device to be unserved must be currently served and not in remote use. Once this command is successfully issued, other client nodes will no longer be able to access the device. A new SERVE command will be needed to return the device back to its served state.

This command is available for network served devices only. SCS devices cannot be unserved.

COMMAND QUALIFIERS　*/ALL*

The ALL qualifier provides you with a simple method of unserving all served devices in a single command. When the ALL qualifier is specified, the "device name" parameter should be omitted.

EXAMPLES

1   `$ DEVICESHARE UNSERVE $2$MUA0`

This example removes the tape device, 2MUA0, from access by other nodes in the network.

3.2.4   Modifying a Served-Device

DeviceShare allows you to modify certain attributes of a served device. You may change the ALIAS names, INCLUDE/EXCLUDE node list and SECURITY level of a served device. All changes are performed in an incremental manner. For example, if you specify the ALIAS qualifier with a single name, that name will be *added* as an alias. To remove an attribute, specify NO in front of the qualifier. For example, NOALIAS will remove all aliases associated with the device (except the device type built-in alias). To change served device attributes, issue the following command:

```
$ DEVICESHARE MODIFY device_name/qualifiers
```

MODIFY

This command will modify a served device's DeviceShare attributes.

FORMAT

DEVICESHARE MODIFY   device_name

Command Qualifiers	Defaults
/[NO]ALIAS[=list]	
/[NO]COMPRESS	
/[NO]EXCLUDE[=list]	
/[NO]INCLUDE[=list]	
/SECURITY=keyword	

PARAMETERS

device_name
This parameter specifies the name of the device about which you want to change DeviceShare attributes.

restrictions

- This command requires OPER and MOUNT privileges.

- The device must be currently served through DeviceShare.

DESCRIPTION

This command will modify a served device's DeviceShare attributes. Attributes which may be modified are: ALIAS, INCLUDE/EXCLUDE node name lists, Network Data Compression and SECURITY level.

COMMAND QUALIFIERS

/[NO]ALIAS[=list]
You may add or remove valid alias names for this NETWORK served device. The device type alias is not affected by this command. To add a new alias simply specify the ALIAS qualifier with one or more new alias names. An alias name cannot exceed fifteen (15) characters. The total number of existing and new alias names (with the exception of the built-in alias) cannot exceed three (3). To remove an alias, negate the ALIAS qualifier (as in NOALIAS) and specify the alias names you wish to remove. Your specification must exactly match an existing alias for it to be removed. To remove all alias names, specify NOALIAS without any list.

/[NO]COMPRESS
This qualifier, valid for NETWORK served devices, allows you to set the default characteristics for network data compression. Network data compression reduces the amount of network traffic by compressing characters which occur at least 4 consecutive times into a three byte sequence. Depending on the content of your data this can significantly increase the throughput and overall speed of your tape operations. Please note that data compression is only provided across the network, data written to the served device is not compressed.

/[NO]EXCLUDE[=list]

You may add or remove nodes which are to excluded from DeviceShare access. To add additional nodes, simply specify the EXCLUDE qualifier and one or more nodenames. Each node entry may use wildcard characters for more flexibility. To remove nodes from the *excluded* list, negate the qualifier (NOEXCLUDE) and specify one or more nodenames to remove. The nodenames must exactly match the exclude list for the node to be removed. To remove all nodes from the excluded list, specify NOEXCLUDE without any nodename list. Please note that a nodename matching string cannot appear exactly on both the INCLUDE and EXCLUDE lists.

/[NO]INCLUDE[=list]

You may add or remove nodes which are to included for DeviceShare access. To add additional nodes, simply specify the INCLUDE qualifier and one or more nodenames. Each node entry may use wildcard characters for more flexibility. To remove nodes from the *included* list, negate the qualifier (NOINCLUDE) and specify one or more nodenames to remove. The nodenames must exactly match the include list for the node to be removed. To remove all nodes from the included list, specify NOINCLUDE without any nodename list. Please note that a nodename matching string cannot appear exactly on both the INCLUDE and EXCLUDE lists.

/SECURITY=keyword

The SECURITY qualifier, observed for *shared* disks, allows you to enforce a level of security for remote users of your disks. Three (3) security levels are available: AUTHENTICATE, SERVER and CLIENT. AUTHENTICATE is the most restrictive form of security and requires the remote user to first authenticate himself prior to accessing the disk. SERVER expects the remote user to have a valid VMS authorization record and the same UIC as the local and remote systems. SERVER does not propagate the user's remote privileges when accessing files on the shared disk. CLIENT is very similiar to SERVER except the user's remote privileges *are* used when accessing files on the shared disk. See Section 3.2.1.3 for more information.

EXAMPLES

1   `$ DEVICESHARE MODIFY $2$MUA0/NOALIAS`

This example removes all alias names (except the device type built-in alias) from the 2MUA0 served device.

3.2.5   Displaying Served-Device Statistics

The DeviceShare utility accumulates various statistics about each device it serves. To display these statistics, issue the following command:

```
$ DEVICESHARE SHOW DEVICE device_name
```

SHOW DEVICE

This command displays information about the served devices(s) that DeviceShare services on this node.

FORMAT

DEVICESHARE SHOW DEVICE   [device_name]

Command Qualifiers	Defaults
/OUTPUT[=filespec]	*/OUTPUT=SYS$OUTPUT*

PARAMETERS

device_name
This parameter specifies the name of the device about which you want information. If you omit this parameter, SHOW DEVICE displays information about all the devices DeviceShare serves.

DESCRIPTION

This command displays information about the served devices(s), serviced by DeviceShare, as follows:

- The device status—available or online

- The total number of I/O operations performed

- The total number of characters transmitted and requested

COMMAND QUALIFIERS

/OUTPUT[=filespec]
/OUTPUT=SYS$OUTPUT (default)
This qualifier represents the name of the file to which you want DeviceShare to direct your output. SYS$OUTPUT is the default file.

EXAMPLES

1 `$ DEVICESHARE SHOW DEVICE`

```
DeviceShare V020-000 Served Devices on SHELBY 17-MAR-1992 11:24:51.26
Copyright (C) 1992 by Advanced Systems Concepts, Inc.  All Rights Reserved.
Licensed to ASCI-INTERNAL

Device: MUA0                      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
```

This example shows statistics for the network served device MUA0:.

2 `$ DEVICESHARE SHOW DEVICE $2$MUA0`

```
DeviceShare V020-000 Served Devices on SHELBY 17-MAR-1992 11:24:51.26
Copyright (C) 1992 by Advanced Systems Concepts, Inc.  All Rights Reserved.
Licensed to ASCI-INTERNAL

Device: $2$MUA0                   Status: In Use      SCS-Served
  ----------I/O Counts----------  --Reads---   --Writes--
  I/O Operations            :        12910         13172
  Characters Transferred    :     49140726      32616692
  Compressed Character Counts :   10097770       6639129
  Compression Effectiveness :  %      20     %      20
```

This example shows statistics for the SCS served tape device, 2MUA0.

3.2.6   Displaying Served-Node Statistics

Note: **This command is applicable to SCS services only.**

The DeviceShare utility accumulates various statistics about each node it serves. To display these statistics, issue the following command:

```
$ DEVICESHARE SHOW NODE node_name
```

SHOW NODE

This command displays information about the node(s) that DeviceShare
services on this node.

FORMAT   **DEVICESHARE SHOW NODE**   **[node_name]**

Command Qualifiers	Defaults
/OUTPUT[=filespec]	/OUTPUT=SYS$OUTPUT

PARAMETERS   *node_name*
This parameter specifies the name of the node about which you want
information. If you omit this parameter, SHOW NODE displays
information about all the nodes DeviceShare services.

DESCRIPTION   This command displays information about the node(s) that DeviceShare
services, as follows:

- The number of requests, currently being processed

- The maximum number of requests ever processed simulataneously

- The total number of characters transmitted and received

- The running total of I/O operations performed

COMMAND   */OUTPUT[=file_spec]*
QUALIFIERS   */OUTPUT=SYS$OUTPUT (default)*
This qualifier represents the name of the file to which you want
DeviceShare to direct your output. SYS$OUTPUT is the default file.

EXAMPLES

1 $ **DEVICESHARE SHOW NODE GEOFFREY**

```
DeviceShare V020-000 Node Statistics on SHELBY 19-SEP-1990 17:24:01.57
Copyright (C) 1990 by Advanced Systems Concepts, Inc.  All Rights Reserved.

DeviceShare Statistics for Node GEOFFREY

Req Queued:         0  Max Queued:             2
Chars Read:   16705488  Chars Written:         0

Operations Counts

ABORT            0   GET UNT STS     71   SET CTL CHR      2
AVAILABLE        8   ONLINE          32   SET UNT CHR     20
ERASE GAP        0   CMP HST DAT      0   REPOSITION      34
READ          2067   WRITE            0   WRT TAPE MK      0
FLUSH            0   GET CMD STS      0
                    Total Operations   2234
```

This example displays statistics for the node, GEOFFREY.

2 $ **DEVICESHARE SHOW NODE**

```
DeviceShare V020-000 Node Statistics on SHELBY 19-SEP-1990 17:24:14.90
Copyright (C) 1990 by Advanced Systems Concepts, Inc.  All Rights Reserved.

DeviceShare Statistics for Node AMANDA

Req Queued:         0   Max Queued:             0
Chars Read:         0   Chars Written:          0

Operations Counts

ABORT            0   GET UNT STS     25   SET CTL CHR      2
AVAILABLE        0   ONLINE           0   SET UNT CHR      0
ERASE GAP        0   CMP HST DAT      0   REPOSITION       0
READ             0   WRITE            0   WRT TAPE MK      0
FLUSH            0   GET CMD STS      0
                    Total Operations     27

DeviceShare Statistics for Node GEOFFREY

Req Queued:         0   Max Queued:             2
Chars Read:   16705488   Chars Written:         0

Operations Counts

ABORT            0   GET UNT STS     71   SET CTL CHR      2
AVAILABLE        8   ONLINE          32   SET UNT CHR     20
ERASE GAP        0   CMP HST DAT      0   REPOSITION      34
READ          2067   WRITE            0   WRT TAPE MK      0
FLUSH            0   GET CMD STS      0
                    Total Operations   2234
```

This example displays statistics for all the nodes that DeviceShare services.

3.2.7   Displaying Server Statistics

Note: **This command is applicable to NETWORK services only.**

DeviceShare can report on the characteristics and use of each served device. In particular, DeviceShare can report on each device and whether it is currently in use and by whom. To display this data, issue the following command:

```
$ DEVICESHARE SHOW SERVER
```

SHOW SERVER

This command displays information about which device(s) are served, and whether the device is in-use or available.

FORMAT   **DEVICESHARE SHOW SERVER**

Command Qualifiers	Defaults
/FULL	*Brief Listing*
/OUTPUT[=filespec]	*/OUTPUT=SYS$OUTPUT*

DESCRIPTION   This command displays information about served devices. For each served device, a list of that device's ALIASES, INCLUDED and EXCLUDED nodes and whether the device is available or in-use. If the device is in-use, information is displayed indicating the node and user on the remote system that is currently using the device.

COMMAND QUALIFIERS

/FULL
Brief Listing (default)
The qualifier allows you to obtain complete information concerning the served devices on your system. When the FULL qualifier is specified, DeviceShare lists all options, included/excluded nodes and the device's aliases as well as whether the device is in use or free. A brief listing, obtained when the FULL qualifier is omitted, displays each served device on a single line with information concerning the device's availability.

/OUTPUT[=file_spec]
/OUTPUT=SYS$OUTPUT (default)
This qualifier represents the name of the file to which you want DeviceShare to direct your output. SYS$OUTPUT is the default file.

EXAMPLES

1 $ **DEVICESHARE SHOW SERVER**

```
DeviceShare V020-000 Served Devices on SHELBY  17-MAR-1992 11:24:57.95
Copyright (C) 1992 by Advanced Systems Concepts, Inc.  All Rights Reserved.
Licensed to ASCI-INTERNAL

Device          Type            Status
MUA0            TU81            Active
```

This example shows a brief listing of all served devices on this node. The device name, device type and an indication of whether the device is available or in use.

2 ```
$ DEVICESHARE SHOW SERVER/FULL
DeviceShare V020-000 Served Devices on SHELBY 17-MAR-1992 11:25:01.97
Copyright (C) 1992 by Advanced Systems Concepts, Inc. All Rights Reserved.
Licensed to ASCI-INTERNAL

Device: MUA0   Type: TU81   Status: Active
  Options: None
  Aliases: FAST   6250
  User   Node   PID
  PETER   SCOTTY   2280005B
```

This example displays the same information as a brief listing, however, additional information such as various options in effect (i.e. compression), the devices aliases, and any remote user access are also displayed.

## 3.2.8 Monitoring DeviceShare Activity

The DeviceShare MONITOR command monitors and controls DeviceShare activity from a video-oriented device. With DeviceShare MONITOR, you can create windows of various information that DeviceShare updates in real-time. In addition DeviceShare MONITOR allows you to issue SPAWN, HELP, and any valid DeviceShare command. To keep track of DeviceShare activity, enter the command, as follows:

```
$ DEVICESHARE MONITOR [DEVICE | NODE]
```

DEVICE provides a list of served devices. NODE provides a list of client VAXcluster nodes. Figure 3–2 illustrates a monitor DEVICE window.

Figure 3–2   MONITOR DEVICE Window

```
                         ─────DEVICE─────
                    ────I/O Operations────  ──I/O Rates KB/Sec──
Device       Status       Reads    Writes      Read    Write
$1$DIA2      Available      106         0      0.00     0.00
$1$DIA3      Available        1         0      0.00     0.00
MUA0         Available       15         9      0.00     0.00











DEVICESHARE V020-000 Monitor on SULU              20-APR-1992 13:30:11.45
Copyright © 1990, 1992 by Advanced Systems Concepts, Inc. All Rights Reserved
ASCI-INTERNAL
```

3.2.8.1   Monitor Command Syntax

While running DeviceShare MONITOR, your selected window(s) are updated at the default-time interval of three (3) seconds. To enter MONITOR commands, you simply start typing. DeviceShare MONITOR moves the cursor to the bottom of the screen and begins echoing your commands. Table 3–1 lists the commands that you can enter at the MONITOR> prompt. The general format for these commands is, as follows:

```
MONITOR> command [window_name [{ AT or TO } position]]
```

Valid *MONITOR* commands are specified in Table 3–1.

Table 3–1   MONITOR commands

Command	Key	Description
ADD	None	Adds a window
ADVANCE	Next Screen	Scrolls the information within a window forward
BACKUP	Prev Screen	Scrolls the information within a window backward
EXIT	Ctrl/Z	Exits the MONITOR utility
MOVE	Insert Here	Moves a window to a new position on the screen
PASTE	None	Pastes a window to the front
PRINT	None	Prints Screen via VMS Print Services
REFRESH	Ctrl/W	Refreshes the screen
REMOVE	Remove	Removes a window
SET	None	Sets a MONITOR parameter
TOP	Find	Resets window display to logical TOP
UNPASTE	None	Makes a window invisible

Valid *position* commands are specified in Table 3–2. You can specify one or more position commands (for example, R10 C5).

Table 3–2 Position Commands

Position	Description
Hn	Halves (H1, top half; H2, bottom half)
Tn	Thirds (T1, T2, T3)
Qn	Quarters (Q1, Q2, Q3, Q4)
Rn	Row number
Cn	Column number
Sn	Window size (in rows)

3.2.8.2 Monitor HELP and SPAWN Commands

DeviceShare MONITOR provides on-line help on all of its facilities, as well as the DeviceShare product itself. Simply type HELP at the MONITOR prompt.

Figure 3–3 MONITOR On-Line Help

```
HELP

    DeviceShare MONITOR is a real-time interactive facility for viewing data
    on devices or nodes serviced by DeviceShare.  DeviceShare MONITOR offers the
    following facilities:

        1) Real-time status displays.
        2) Ability to issue DeviceShare commands from DeviceShare MONITOR.
        3) SPAWN to DCL.
        4) Online MONITOR HELP.

    DeviceShare MONITOR is executed by DCL command; the format is:

        DEVICESHARE MONITOR window_name

    The ADD command help lists the valid window names.

Topic? █
```

DeviceShare MONITOR also supports the SPAWN DCL command, which allows you to spawn a subprocess in which you can execute other DCL commands. Type SPAWN at the MONITOR prompt to create a subprocess.

MONITOR

This command allows you to monitor and control DeviceShare.

FORMAT

DEVICESHARE MONITOR   [window_name]

Command Qualifiers	Defaults
/INTERVAL=delta-time	/INTERVAL="00:00:03"

PARAMETERS

window_name

This parameter specifies the window name you want to create initially. Valid window names are DEVICE and NODE (SCS only). If you omit this parameter, MONITOR creates the DEVICE window.

DESCRIPTION

The DeviceShare MONITOR command allows you to interactively monitor DeviceShare devices and/or served nodes.

COMMAND QUALIFIERS

/INTERVAL=delta-time

/INTERVAL changes the default-refresh rate at which MONITOR updates the status window(s) with new or changed information. The time, entered, must not include **days**.

EXAMPLES

1 $ `DEVICESHARE MONITOR`

This command produces a DEVICE window that displays devices, served by DeviceShare.

4   USING DEVICESHARE ON THE CLIENT

Note: **This chapter deals specifically with client operations using the NETWORK method of serving devices. The SCS method of serving a tape device is completely transparent in all circumstances. The device will automatically appear on all client members within the VAXcluster without any special manual intervention, once the device has been served.**

Use of DeviceShare on the client system is very simple and straightforward. If you want to use a served device, you will need to know the node and device name (or alias).

4.1   Exclusivity

When you want to use a served device, you need to ALLOCATE the device to your system. Tape devices are by their nature exclusive-use devices. When you use a tape drive, it is available to no one else. Disk drives, however, do allow both exclusive and shared use. One simple rule governs the use of a served disk. If you want to share the served device in a non-exclusive manner (i.e. allow others to also have read/write access to the disk), then the disk must be mounted system-wide on the server system. This allows all client systems as well as the server system to have read/write access to the disk. If you want to access a served disk in an exclusive manner then the served disk must not be mounted as a Files-11 volume on the server. This safeguards against inadvertant or accidental destruction of the volume. Please note that the terms *exclusive* and *non-exclusive/shared* apply to nodes not users. If a disk is accessed in a shared manner then all users on your node may access that disk, assuming appropriate VMS privileges.

When a served tape device is allocated, the device must still be mounted prior to use. The tape drive can be mounted either as an ANSI labeled tape or foreign. Commands you issue to the tape drive are standard VMS commands as if the tape device were local to the system. Allocation of the served tape device merely reserves the tape device for your exclusive use and establishes the communications path for subsequent tape I/O.

When a served disk device is allocated exclusively, the device must be mounted *locally* (either Files-11 or foreign) for you to access it. If a served disk is allocated shared, the device will be automatically mounted system-wide as a Files-11 volume.

4.2 Allocating a Served Tape Device

To access a served device you issue the DeviceShare command ALLOCATE. This command establishes a communications path for the serving node, and determines whether you are allowed to access the device and if the device is available for your type of access.

```
$ DEVICESHARE ALLOCATE VAXA::$2$MUA0: MYTAPE
```

The above command allocates the served tape device on VAXA named 2MUA0: and assigns the logical MYTAPE to a local pseudo device which represents, for the life of this connection, the served device. If another user had the tape drive allocated, an error would have resulted on this allocation.

Another variation of the ALLOCATE command is the use of the generic ALIAS naming feature. For example, suppose VAXA had two TU81 tape drives both capable of supporting 6250bpi tape density (2MUA0 and 2MUA1). The system manager for the serving node could have established an alias of 6250BPI for *both* tape drives. Now if 2MUA0: is currently in use, and you issue:

```
$ DEVICESHARE ALLOCATE VAXA::6250BPI MYTAPE
```

DeviceShare would allocate 2MUA1 because it had the same alias name and was available for use at the time. Of course, alias names can be different and might be used as a form of shorthand or memory-aid for describing a tape. Since most systems have only one (1) tape drive, a network convention on the use of the alias TAPE would make it simple for any user in the network to use a tape drive on any other node.

DeviceShare also supports the use of network data compression to potentially improve tape throughput and network utilization. This feature, normally disabled by default, can be enabled on either the SERVE or ALLOCATE command.

Note: **The SERVE command establishes the default network data compression for subsequent ALLOCATE commands. The ALLOCATE command can, however, override the default.**

4.3   Allocating a Served Disk Device

As stated in the previous sections, you may allocate a disk device in a shared manner if the disk device is mounted as a Files-11 volume (system-wide). Three types of disk allocations are possible through DeviceShare.

4.3.1   Exclusive Disk

You may allocate a served disk exclusively provided:

1   You are appropriately privileged

2   No one else has the drive mounted (in any fashion) and no open channels are present

Once the drive is allocated to your system you may issue any VMS command (including INITIALIZE and MOUNT) and all VMS Disk I/O operations are available with complete lock management.

The command:

```
$ DEVICESHARE ALLOCATE VAXA::$1$DUA10: MYDISK
```

will allocate the drive 1DUA10 on VAXA to this system exclusively. You would normally mount the disk (either Files-11 or Foreign) after the ALLOCATE.

4.3.2   Private Shared Disk Access

DeviceShare allows a user to gain access to a served disk for the purposes of sharing information with other nodes, while using the privileges and identifiers of the remote user. This method is used when the disk is to be mounted on the client system in a private fashion, while all I/O requests sent to the server system are evaluated with the user's server security profile as it exists on the VAX (i.e. SYSUAF.DAT). This is performed with the following command:

```
$ DEVICESHARE ALLOCATE/SHARE VAXA"USERNAME PASSWORD"::$1$DUA10: MYDISK
```

The allocation will verify that 1DUA10: is mounted Files-11 (system-wide) on VAXA. Your username and password (optional, if proxy access is enabled) will be verified on VAXA as a legal and valid user. If the allocation is successful, 1DUA10 is mounted *privately* on your system and is available through the logical name *MYDISK*. The logical name

MYDISK is placed in the user's JOB logical name table. All subsequent I/O operations will use the privileges, identifiers and disk quotas of USERNAME as it exists on the server system.

4.3.3   System-wide Shared Disk Access

DeviceShare also allows multiple users to gain access to a shared served disk. The command is:

```
$ DEVICESHARE ALLOCATE/SHARE/SYSTEM VAXA::$1$DUA10: OURDISK
```

The SYSTEM qualifier indicates that the allocation should be made available to the system. No DECnet access control information is specified with the node name. Assuming 1DUA10 is mounted as a Files-11 volume (system-wide), that volume will be mounted on the client node, system-wide, and accessed through the logical name *OURDISK*. The logical name OURDISK is placed in the SYSTEM logical name table. The VMS System Manager on the Server system can enforce one of three levels of shared disk security.

1   **Authenticate**

This level of security indicates that you must first use the DeviceShare AUTHENTICATE command in order to validate your access rights to the shared disk. You must have a valid VMS username/password on the server system before you can use the disk device. Any attempt to access an *authenticate* shared disk without authorization results in an SS$_NOPRIV error.

2   **Server**

This level of security assumes that your username and UIC on the remote system match that of the server system. If so, your privileges, identifiers and disk quotas from the server system are used whenever you attempt to access the shared served disk. If not, an SS$_NOPRIV error is passed back.

3   **Client**

This level of security uses your remote UIC and privileges whenever I/O operations are performed.

ALLOCATE

This command establishes a communications path to the served device on the remote node and creates a local pseudo device through which I/O operations are issued.

FORMAT

DEVICESHARE ALLOCATE served_device
logical_name

Command Qualifiers	Defaults
/[NO]COMPRESS	/NOCOMPRESS
/[NO]LOG	/NOLOG
/NAME=process-name	
/[NO]SHARE	/NOSHARE
/SYSTEM	Job Logical Name Table

prompts

Served-device: served_device
Logical-name: logical_name

restrictions

- MOUNT privilege is required in all cases.

- SYSNAM, VOLPRO and OPER is required for exclusive and system-wide shared device allocation.

- Shared disk devices cannot be dismounted through VMS. You must use the DeviceShare DEALLOCATE command.

- DeviceShare will force FIB$M_NOWRITE to ensure that shared writing is not allowed. Also, do not INSTALL images from the remote disk on your local system.

- DeviceShare devices cannot be MSCP served.

PARAMETERS

served_device
Specifies the server node name and the served device name (or alias) in the form of node_name::served_device_name. The node_name should contain access information for a private shared disk allocation.

logical_name
Specifies the logical name that is to be associated with the pseudo device created as part of the ALLOCATE command. All subsequent device usage should reference this logical name.

DESCRIPTION This command allocates a served device for your use. The allocation can be exclusive or shared depending on the attributes of the device. A successful exclusive allocation requires that the served device be mounted for further use. A successful private shared allocation results in the disk device mounted privately as a Files-11 volume. A successful system-wide shared allocation results in the disk device mounted system-wide as a Files-11 volume. While shared served disks can be written to, a file on that disk cannot be opened for write sharing. DeviceShare will transparently disable write sharing for a file to be opened in this mode.

The logical name specified with this command is created in the LNM$JOB table for a private shared disk and in the LNM$SYSTEM table for an exclusive or system-wide shared disk allocation. ASCI recommends that all subsequent device usage refer to this logical name.

DeviceShare will set device protection to (S:RWED,O:RWED,G:,W:) for tape devices, (S:RWED,O:RWED,G:,W:) for private shared disk devices and (S:RWED,O:RWED,G:RWED,W:RWED) for system-wide shared disk devices.

Please note that the VMS SHOW DEVICE command will not correctly display the number of free disk blocks or other Files-11 dynamic information for a served shared disk.

COMMAND QUALIFIERS **/[NO]COMPRESS**
/NOCOMPRESS (default)
This qualifier, allows you to override the default characteristics for network data compression. Network data compression reduces the amount of network traffic by compressing characters which occur at least 4 consecutive times into a three byte sequence. Depending on the content of your data this can significantly increase the throughput and overall speed of your tape operations. Please note that data compression is only provided across the network, data written to the served device is not compressed.

/[NO]LOG
/NOLOG (default)
This qualifier will cause successful allocations to display a message indicating more information about the operation.

/NAME=process-name
This qualifier allows you to name the detached task which is started when you sucessfully complete an ALLOCATE command. By default, DeviceShare attempts to name the process as "node$device-name". If the resultant string is greater than fifteen (15) characters, DeviceShare names the process "DEVSHR_nnnn" where "nnnn" is a number starting at 1 and continuing in ascending sequence. If you specify the NAME qualifier, the value string must be a valid VMS process name specification not exceeding fifteen (15) characters.

/[NO]SHARE
/NOSHARE (default)
This qualifier determines whether the device is to allocated exclusively to this node or in a shared manner with respect to other nodes. Tape

devices are always allocated exclusively and this qualifier is ignored. If a disk device is to allocated exclusively (NOSHARE) then it cannot be mounted, in any manner, on the serving system. Likewise, if the disk is to allocated in a shared manner (SHARE) then it must be mounted as a system-wide Files-11 ODS-2 volume on the serving system. The absence or presence of the SYSTEM qualifier determines whether the allocation is made system-wide or privately. Exclusive disks provide complete range of I/O operations with no restriction. Shared disks allow both read and/or write operations, however, files are always opened for read sharing only when accessed through DeviceShare. This means that applications which expect to perform write sharing file opens can expect a "File is locked" if another process and/or channel has the file open for write sharing.

File write sharing is not supported through DeviceShare since VMS lock management across nodes requires VAXcluster operation.

Exclusive disk logical names are placed in the SYSTEM logical name table.

/SYSTEM
Job Logical Name Table (default)
This qualifier when specified together with the SHARE qualifier determine whether the shared disk allocation should be performed system-wide. If SYSTEM is omitted, access control information **must** be specified with the node name. If SYSTEM is specified, access control information must be omitted. By default, shared disk devices are allocated privately.

EXAMPLES

1   `$ DEVICESHARE ALLOCATE VAXA::MKA500 TAPE`

This example allocates the TK50 drive on VAXA for exclusive use.

2   `$ DEVICESHARE ALLOCATE VAXA"DEMO"::$1$DUA1: DEVDISK`

This example allocates the disk drive 1DUA1 on node VAXA for exclusive use. To continue to use the disk drive, the MOUNT command must be issued.

3   `$ DEVICESHARE ALLOCATE VAXA::$1$DUA10:/SHARE/SYSTEM DEVDISK`

This example allocates the disk drive 1DUA10 on node VAXA for shared use. The disk will be mounted by DeviceShare for system-wide access.

4.4 Authenticating Access to a Shared Disk

The Server system manager can designate a shared disk device's security level. When the level is set to *authenticate*, you must have a valid VMS username and password on the server system prior to access the remote disk. For example,

```
$ DEVICESHARE AUTHENTICATE OURDISK
```

The above command enters an authentication sequence for the remote disk *OURDISK*. By default, the username is assumed to be the current username on the local system. If this is not true, specify the /USERNAME qualifier with the AUTHENTICATE command. By default, DeviceShare will prompt for your *server* password which matches the server VMS account for that username. The password prompt is not echoed. If your network and server system support network proxy access, you may press Return . To avoid the prompt, you may also specify either NOPASSWORD or PASSWORD=password along with the AUTHENTICATE command.

The AUTHENTICATE command can also be used for a security level of *server* to remove abiguities concerning username/UIC mismatches. For example, user STEVE (UIC [20,1] on node A) wants to access node B's disk. Node B's STEVE record is associated with UIC [22,5]. Normally, DeviceShare would pass back an SS$_NOPRIV error since the UIC's don't match. However, if STEVE on Node A executes a valid authentication sequence to Node B, Node B will use the proper security record and continue processing I/O operations as expected.

The AUTHENTICATE command cannot be used for *client* level security since all access is permitted in any event.

AUTHENTICATE

This command verifies a remote user's access to a system shared disk.

FORMAT   **DEVICESHARE AUTHENTICATE   logical-name**

Command Qualifiers	Defaults
/[NO]PASSWORD[=string]	/PASSWORD
/USERNAME[=string]	/USERNAME=current_username

prompts   Logical-Name: logical_name

PARAMETERS   *logical_name*
Please specify the same logical name you associated with the served device as part of the ALLOCATE command.

restrictions   •   AUTHENTICATE cannot be used for *client* level security shared disks.

DESCRIPTION   A system shared disk may be designated by the server system as requiring authentication prior to access. This means that you must have a valid VMS username account on the server system before you can access the shared disk. If you receive the message, "Not on Creator Port", you must be a valid VMS user of the server system before you can gain access. The authentication sequence will use the value from the USERNAME qualifier (your local username, if not provided) and the PASSWORD qualifier (your password will be requested if not specified) and attempt to validate you as a valid VMS user on the server system. A password is not required if proxy access is enabled.

COMMAND QUALIFIERS   */[NO]PASSWORD[=string]*
/PASSWORD (default)
This qualifier determines whether DeviceShare will prompt you for a server remote password. If specified as NOPASSWORD, DeviceShare will not prompt for a password. This means that proxy access must be enabled or your authentication sequence will fail. If the qualifier is omitted, DeviceShare will prompt for a password without echo. The default action is for DeviceShare to prompt you for a password.

/USERNAME[=string]
This qualifier allows you to enter a different username than the one who have logged in as on the client system. If the USERNAME qualifier is specified, you must enter a password. If the USERNAME qualifier is omitted, DeviceShare will use the current client username.

EXAMPLES

1 $ `DEVICESHARE AUTHENTICATE OURDISK`

This example requests an authentication sequence for the current username. The user's password is prompted separately.

4.5   Deallocating a Served Device

When you are finished with a served device you use the DEALLOCATE command to make it available for other nodes and users. If the device has been allocated exclusively, you must dismount the device prior to issuing the DEALLOCATE. If the device has been allocated shared, you must ensure that no channels are currently referencing the device. DeviceShare will check that no one is using a shared served device before dismounting the disk as part of the DEALLOCATE command.

DEALLOCATE

This command releases a served device to other nodes and users.

FORMAT

DEVICESHARE DEALLOCATE   logical-name

Command Qualifiers	Defaults
/[NO]LOG	/NOLOG

prompts   Logical-name: logical_name

PARAMETERS

logical_name
Please specify the same logical name you associated with the served device as part of the ALLOCATE command.

restrictions

- MOUNT privilege is required in all cases.

- SYSNAM, VOLPRO and OPER is required for exclusive and system-wide shared device allocation.

DESCRIPTION

This command deallocates a served device and makes it available for other nodes and users. If the device is a shared served device, the device will be dismounted automatically as part of the DEALLOCATE command.

COMMAND QUALIFIERS

/[NO]LOG
/NOLOG (default)
This qualifier will cause successful allocations to display a message indicating more information about the operation.

EXAMPLES

1 $ `DEVICESHARE DEALLOCATE MYTK50`

This example deallocates the served device represented by the logical name MYTK50.

4.6   Communicating with a Remote Operator

DeviceShare provides a command which allows you to use OPCOM
facilities remotely. This command enables you to tell a remote operator
what tape or disk to mount, and to receive a reply, just as if your mount
request were local. The OPCOM command is used to send a message to
a remote operator concerning a served device. The served device must
already be ALLOCATEd to the system. You can indicate which OPCOM
class the message is to be directed to and you can receive a reply to your
message. The qualifier /REPLY indicates that you want to capture the
operator's reply. The REPLY qualifier requires a DCL symbol-name into
which the text of the reply is stored. The symbol-name does not have to
previously exist. If it does, it must be a string datatype. The use of the
DCL symbol makes analysis of the reply simple within a DCL command
procedure. For example, you might want to ask the operator whether
backups should be done at this time. The operator's reply can then be
programmatically analyzed as to whether you should proceed with the
backup.

```
$ DEVICESHARE ALLOCATE MYNODE::$2$MUA0: TAPE
$ DEVICESHARE OPCOM /REPLY=ANSWER TAPE "Please load tape BCK912."
$ IF ANSWER .EQS. "OK" THEN GOTO BACKUP_DISK
$ IF ANSWER .EQS. "NG" THEN GOTO ABORT_BACKUP
$BACKUP_DISK:
$ MOUNT/FOR TAPE:
$ BACKUP/IMAGE local_disk TAPE:BCK912/save
$ DISMOUNT TAPE:
$ DEVICESHARE OPCOM TAPE "Please remove tape BCK912. Thank you."
$ABORT_BACKUP:
$ DEVICESHARE DEALLOCATE TAPE
$ EXIT
```

This example demonstrates a sample command procedure which might be
used to acquire a remote tape drive and perform a local disk backup. The
command procedure is simplistic in that no error handling logic has been
written. The first command allocates the remote tape drive. The logical
name TAPE represents the remote device's *alter ego* on this node. The
logical name TAPE is then used for all subsequent VMS commands. The
tape is mounted foreign, and a backup is started from the local disk to the
remote tape. When the backup has ended, the tape is dismounted. The
OPCOM command is used initially to request the mounting of a specific
tape as well as asking permission for the backup to commence. Please note
the use of the REPLY qualifier. The symbol ANSWER will be used to store
the remote operator's answer to our message. We can then inspect his
answer to determine whether he complied with our request. The second
use of the OPCOM command shows a simple message directed to the
operator with no reply expected.

OPCOM

This command sends a message to a remote operator of a served device. The operator's reply, if requested, can be retrieved and made available for analysis.

FORMAT   **DEVICESHARE OPCOM   logical-name message**

Command Qualifiers	Defaults
/CLASS=classes	/CLASS=CENTRAL
/REPLY[=symbol]	No reply

prompts   Logical-name: logical_name

PARAMETERS   ***logical_name***
You specify the logical name used in the ALLOCATE command

message
Specifies the message string, enclosed within quotes, that is to be sent to the remote operator

restrictions
- MOUNT privilege is required in all cases.

- SYSNAM, VOLPRO and OPER is required for exclusive and system-wide shared device allocation.

DESCRIPTION   This command is used to send messages to remote operator concerning served devices or other operations concerning the remote node. The message is displayed via OPCOM facilities and, by default, is sent to the Central operator class. The CLASS qualifier allows you to direct the message to other operator classes. If a request type message is desired, use of the REPLY qualifier causes OPCOM to query the remote operator for a reply to the message. The operator's reply is always displayed on SYS$OUTPUT of the issuing process. If the reply is to be analyzed by a program or procedure, a DCL symbol-name can be specified with the REPLY qualifier. The reply is then also stored within this symbol where it can be examined. When a reply is requested, the OPCOM command will wait indefinitely for the reply. OPCOM on the remote system will re-query the operator at least every two (2) minutes requesting an answer.

COMMAND   ***/CLASS=list***
QUALIFIERS   ***/CLASS=CENTRAL (default)***
This qualifier will direct the OPCOM message to one or more OPCOM classes. By default, the CENTRAL operator is used.

/REPLY[=symbol]

This qualifier indicates that the message should be sent as request. The remote operator is directed to answer the query, which is displayed on SYS$OUTPUT of the process issuing the OPCOM command. If you need to analyze the reply programmatically, a DCL symbol-name should be specified. The symbol does not have to exist, but if it does it must be of string datatype. The symbol-name specified must obey DCL symbol naming rules.

EXAMPLES

1   `$ DEVICESHARE OPCOM MYTK50 "Please mount TK50 named SAMPLE"`

This example sends a message to the operator represented by the logical name MYTK50 (original allocation VAXA::MKA500). The operator is asked to mount a TK50 labeled SAMPLE.

2   `$ DEVICESHARE OPCOM/REPLY=ANSWER MYTK50 "Please mount TK50 named SAMPLE"`

This example sends a message to the operator represented by the logical name MYTK50 (original allocation VAXA::MKA500). The operator is asked to mount a TK50 labeled SAMPLE. This time the operator is asked to reply to the request. The reply is stored in the DCL symbol ANSWER as well as displayed on SYS$OUTPUT.

A MESSAGES

This section describes all of the known condition codes that DeviceShare can display, while in use.

NOTE: A secondary message that further explains the primary message you might encounter accompanies many DeviceShare messages. Although ASCI has made every attempt to document the system errors which can occur, the secondary-error code can refer to almost any VMS error. However, in most cases, the error is self-explanatory.

A.1 MESSAGES

ALEXCMAX,   number of aliases exceeds maximum permitted

Facility: DeviceShare

Severity: Error

Explanation: Self-explanatory.

User Action: Correct and retry.

ALIASERR,   error occurred modifying alias string

Facility: DeviceShare

Severity: Error

Explanation: This message is accompanied by another more explanatory message.

User Action: Correct and retry.

ALRALLOC,   Device Already Allocated

Facility: DeviceShare

Severity: Error

Explanation: You attempted to allocate a device which is already in use.

User Action: Correct and retry.

BADSDDB,   device block is invalid

Facility: DeviceShare

Severity: Error

Explanation: This error should not be displayed and indicates a product inconsistency.

User Action: If you are under Warranty and/or maintenance services, please contact ASCI or your local distributor.

BADSECHDR,   section header is invalid

Facility: DeviceShare

Severity: Error

Explanation: This error should not be displayed and indicates a product inconsistency.

User Action: If you are under Warranty and/or maintenance services, please contact ASCI or your local distributor.

CLIERR,   error received from CLI function

Facility: DeviceShare

Severity: Error

Explanation: An error occurred, when DeviceShare attempted to retrieve command-line information. A secondary message that explains the problem in more detail accompanies this message.

User Action: Based upon the secondary message, ascertain the cause of the problem, and correct it.

CONNTMO,   reconnection attempt timed out at *date-time*

Facility: DeviceShare

Severity: Error

Explanation: This message is displayed when a network link failure occurs and the time to wait and attempt reconnection has expired. The link is marked as down.

User Action: All processes and programs using the remote device should be aborted. The device should then be deallocated.

DEVACT,   device is currently active

    Facility: DeviceShare

    Severity: Error

    Explanation: DeviceShare displays this message, when attempting to serve a tape device currently in use.

    User Action: Ensure that no one is using the tape device. Then retry the command.

DEVALLOC,   Device *string* successfully allocated

    Facility: DeviceShare

    Severity: Successful

    Explanation: The device you selected has now been allocated.

    User Action: None.

DEVDEALLOC,   Device *string* successfully deallocated

    Facility: DeviceShare

    Severity: Successful

    Explanation: The device you selected has now been deallocated.

    User Action: None.

DEVMNTD,   device is mounted

    Facility: DeviceShare

    Severity: Error

    Explanation: You attempted to ALLOCATE a device exclusively, however, the disk device is already mounted as a Files-11 volume.

    User Action: Either use a different device or try again later.

DEVNOTAVL,   device not available

    Facility: DeviceShare

    Severity: Error

    Explanation: You attempted to ALLOCATE a device which is currently in use.

    User Action: Either use a different device or try again later.

DEVSRV,   device is already served

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** DeviceShare attempted to serve a tape device that it was already serving.
>
> **User Action:** None

GDIERR,   error getting device information

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** An error occurred in processing a SHOW DEVICE command. A secondary message that explains the problem in more detail accompanies this message.
>
> **User Action:** Based upon the secondary message, ascertain the cause of the problem, and correct it.

GNIERR,   error getting node information

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** An error occurred in processing a SHOW NODE command. A secondary message that explains the problem in more detail accompanies this message.
>
> **User Action:** Based upon the secondary message, ascertain the cause of the problem, and correct it.

INSFMEM,   internal database is full

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** You attempted to SERVE more devices when DeviceShare was configured for.
>
> **User Action:** If you are under warranty or maintenance services, contact ASCI (or your local distributor) for additional assistance.

INVBLK,   invalid block in queue

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** An internal-DeviceShare error occurred.
>
> **User Action:** If you are under warranty or maintenance services, contact ASCI (or your local distributor) for additional assistance.

INVDEV,   device is not a tape or disk

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** You attempted to issue a DeviceShare command to a device which is neither tape or disk.
>
> **User Action:** Please check the device type.

INVMSG,   invalid message received: *code*

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** This message should never be displayed.
>
> **User Action:** If you are under warranty or maintenance services, contact ASCI (or your local distributor) for additional assistance.

INVWIND,   invalid window name specified

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** You issued a MONITOR command and did not reference the DEVICE or NODE window names. Anyother window names are invalid.
>
> **User Action:** Correct and try again.

LINKDOWN,   link down condition detected at *date-time*

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** This message indicates that a network link failure occurred.
>
> **User Action:** Please investigate the cause of the network link failure. If the failure is temporary, DeviceShare will automatically resume operations.

MEMALCERR,   unable to obtain dynamic memory

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** This error indicates that a request to LIB$GET_VM failed.
>
> **User Action:** Usually this is correctable by increasing either the process's PGFILCNT quota or VIRTUALPAGES SYSGEN parameter.

NETSERVERR,   unable to serve device

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** DeviceShare was unable to SERVE the device you specified. This message is normally accompanied by an additional message which provides more details.
>
> **User Action:** Correct and try again.

NODERR,   error occurred modifying node name entry

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** This message is accompanied by another more explanatory message.
>
> **User Action:** Correct and retry.

NODRIVER,   DeviceShare device driver not loaded

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** The DeviceShare-device driver was not loaded, when you issued a DeviceShare command.
>
> **User Action:**
>
> **1** Verify that you started the DeviceShare utility.
>
> **2** Resubmit the command.

NOSUCHALIAS,   alias string not found

   Facility: DeviceShare

   Severity: Error

   Explanation: DeviceShare couldn't find the device you asked for by alias.

   User Action: Correct and retry.

NOSUCHNODE,   node block not found

   Facility: DeviceShare

   Severity: Error

   Explanation: Normally this is a message which should never be encountered.

   User Action: If you are under warranty or maintenance services, contact ASCI (or your local distributor) for additional assistance.

NOSUCHSEC,   section not loaded

   Facility: DeviceShare

   Severity: Error

   Explanation: Normally this is a message which should never be encountered.

   User Action: Please check that DeviceShare was started. If you are under warranty or maintenance services, contact ASCI (or your local distributor) for additional assistance.

NOTFILES11,   device is not mounted Files-11

   Facility: DeviceShare

   Severity: Error

   Explanation: You attempted to allocate a disk device as shared bu the target disk is not mounted as a Files-11 volume, system-wide.

   User Action: Please contact the system manager of the remote system and ensure the disk is mounted Files-11, system-wide.

NOTSERVER,   no devices are currently being served

   Facility: DeviceShare

   Severity: Error

   Explanation: There are no served devices on the system.

   User Action: None

OUTPERR,   error opening output file

Facility: DeviceShare

Severity: Error

Explanation: An error occurred, when DeviceShare attempted to open an output file. A secondary message that explains the problem in more detail accompanies this message.

User Action: Based upon the secondary message, ascertain the cause of the problem, and correct it.

REPLY,   *remote-operator-reply*

Facility: DeviceShare

Severity: Informational

Explanation: This message is printed when the remote operator responds to a request via the OPCOM command. The text displayed is the answer to your request.

User Action: None.

RESTRTSUC,   link restart successful at *date-time*

Facility: DeviceShare

Severity: Successful

Explanation: DeviceShare automatically restarted the network link after encountering a failure.

User Action: None.

RQSTABORT,   *string*, Request *number* was aborted by operator *name*

Facility: DeviceShare

Severity: Error

Explanation: An OPCOM request was aborted by the target operator.

User Action: None.

RQSTCAN,   *string*, Request *number* was cancelled

Facility: DeviceShare

Severity: Error

Explanation: An OPCOM request was cancelled by the target operator.

User Action: None.

RQSTCMPLTE, *string*, Request *number* was completed by operator *name*

> **Facility:** DeviceShare
>
> **Severity:** Successful
>
> **Explanation:** An OPCOM request was completed by the target operator.
>
> **User Action:** None.

TNSINIERR, error initiating DEVICESHARE Network Server

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** An error occured starting the network server.
>
> **User Action:** If you are under warranty or maintenance services, contact ASCI (or your local distributor) for additional assistance.

TNSSEQNOERR, Sequence Number mismatch - expecting *number*, received *number*

> **Facility:** DeviceShare
>
> **Severity:** Error
>
> **Explanation:** All message sent by DeviceShare are sequence numbered to ensure proper transmission and reception of all data. If the expected sequence numbers do not match, this error message is produced and the link is terminated.
>
> **User Action:** If you are under warranty or maintenance services, contact ASCI (or your local distributor) for additional assistance.

TNSSTART, DEVICESHARE Network Server started at *date-time*

> **Facility:** DeviceShare
>
> **Severity:** Informational
>
> **Explanation:** This message is printed when the Network Server for a device is initiated.
>
> **User Action:** None.

VERSMISMAT,   version mismatch

Facility: DeviceShare

Severity: Error

Explanation: This message indicates that components of the DeviceShare system are not at the same version level.

User Action: If you have just updated the system with a new version of DeviceShare, you must reboot to continue proper operation. Otherwise, you should re-install the product.

WRGSTATE,   message received in wrong state: *code*

Facility: DeviceShare

Severity: Error

Explanation: Normally this is a message which should never be encountered.

User Action: If you are under warranty or maintenance services, contact ASCI (or your local distributor) for additional assistance.

GLOSSARY

This is a page for the dummy glossary.

Print the glossary with the following command:

document/batch=(noprint,keep,notify)/print vir_gloss.sdml report.twocol ln03

Index

Index

M

N

O

P

R

S