

---

# DECram for OpenVMS User's Manual

**April 2001**

This manual describes the features of the DECram for OpenVMS Device Driver (MDDRIVER). It includes information on how to determine which files should be stored on the device, how to configure the device, and how to use the device driver.

<b>Revision/Update Information</b>	This is a new document
<b>Operating System and Version:</b>	OpenVMS Alpha Version 7.2-1H1 or higher
<b>Software Version:</b>	DECram for OpenVMS Version 3.0

**Compaq Computer Corporation  
Houston, Texas**

---

© 2001 Compaq Computer Corporation

Compaq, VAX, VMS, and the Compaq logo Registered in U.S. Patent and Trademark Office.

All other product names mentioned herein may be the trademarks or registered trademarks of their respective companies.

Confidential computer software. Valid license from Compaq required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Compaq shall not be liable for technical or editorial errors or omissions contained herein.

The information in this document is provided "as is" without warranty of any kind and is subject to change without notice. The warranties for Compaq products are set forth in the express limited warranty statements accompanying such products. Nothing herein should be construed as constituting an additional warranty.

ZK5698

The DECram documentation set is available on CD-ROM.

This document was prepared using DECdocument, Version 3.3-1b.

---

# Contents

<b>Preface</b> .....	v
<b>1 DECram Disks</b>	
1.1 Characteristics of a DECram Disk .....	1-1
1.1.1 Comparing a DECram Disk to a Conventional Disk .....	1-2
1.1.2 Comparing a DECram Disk to Disk Caches .....	1-2
<b>2 DECram Disk Configuration</b>	
2.1 Preparing to Configure a DECram Disk .....	2-1
2.1.1 Resource Requirements .....	2-2
2.2 Configuring DECram Disks .....	2-2
2.2.1 Creating a DECram Disk .....	2-3
2.2.1.1 OpenVMS Alpha All VMS Versions to Create Local Disks .....	2-3
2.2.1.2 OpenVMS Alpha Version 7.2-1H1 and Later: Using DECram Version 3.0 (or Later) to Create Local and Shared Disks .....	2-4
2.2.2 Allocating and Deallocating Space for a DECram Disk .....	2-4
2.2.2.1 Allocating and Deallocating Space Using INITIALIZE .....	2-4
2.2.2.2 Allocating and Deallocating Space Using DECram Version 3.0 and Later .....	2-5
2.2.2.2.1 Special Notes for Shared Disks During Allocation and Deallocation .....	2-8
2.2.3 Mounting a DECram Disk .....	2-8
2.2.4 Mounting a Local DECram Disk .....	2-8
2.2.5 Mounting a Shared DECram Disk .....	2-9
2.3 Restoring an OpenVMS Alpha Version 7.2-1H1 (or Later) DECram Disk .....	2-10
2.3.1 Restoring a Disk Using SYSMAN STARTUP .....	2-10
2.3.2 Restoring a Disk Using the DCL command DECGRAM RECOVER .....	2-10
2.4 Determining Allocation of DECram Device Resources .....	2-11
2.4.1 Determining Resource Allocation Using DECram Version 3.0 and Later .....	2-11
2.5 Shadowing a DECram Disk .....	2-12
2.5.1 Mounting a Shadowed DECram Disk–DECram Disk to DECram Disk .....	2-13
2.5.2 Mounting a Shadowed DECram Disk–DECram Disk to Physical Disk or Partition .....	2-14
2.5.2.1 Creating the DECram Disk .....	2-14
2.5.2.2 Mounting a Shadowed DECram Disk .....	2-15

### 3 Using a DECram Disk

3.1	Recommended Applications for a DECram Disk .....	3-1
3.1.1	Identifying Commonly Used Images .....	3-2
3.1.2	Using DECram Disks with a Workstation .....	3-2
3.1.3	Storing SYSSCRATCH Files on a DECram Disk .....	3-2
3.1.4	Recovering Writable DECram Files .....	3-3
3.2	Using a Search List to Locate Files .....	3-4
3.3	Shadow Sets of DECram Devices .....	3-4
3.4	DECram Limitations .....	3-5
3.4.1	Impact to Paging and Swapping .....	3-5
3.4.2	I/O Delay Characteristics Affecting DECram Use .....	3-5
3.4.3	Implications of Local Served and Shadowed DECram Devices .....	3-5
3.4.3.1	Enhanced Operation .....	3-6

### Index

#### Tables

1-1	DECram Disk Versus Conventional Disk .....	1-2
1-2	DECram Disk Versus Disk Caches .....	1-3
2-1	Creating a DECram Disk .....	2-3
3-1	Sources of Temporary Files .....	3-3

---

# Preface

This manual describes the DECram for OpenVMS Device Driver (MDDRIVER). You will find information on how to determine which files should be stored on the device, how to configure the device, and how to use the device driver.

## Intended Audience

This manual is intended for system managers who want to take advantage of the increased performance that results from using the DECram driver. You should be familiar with managing the OpenVMS Alpha operating systems before reading this manual.

## Document Structure

This manual consists of three chapters only as follows:

- Chapter 1 describes the DECram disk and compares it to a conventional disk and to a cached disk.
- Chapter 2 describes how to configure a DECram disk.
- Chapter 3 provides information on the recommended use of a DECram disk.

## Related Documents

See the following documents for information that is relevant to configuring and programming DECram disks:

- *OpenVMS I/O User's Reference Manual*
- *OpenVMS Programming Concepts Manual*
- *Creating an OpenVMS Alpha Device Driver from an OpenVMS VAX Device Driver*
- *Writing OpenVMS Alpha Device Drivers in C*
- *OpenVMS User's Manual*

The following documents contain information that will assist you in identifying files that can be stored on a DECram disk:

- *OpenVMS Performance Management*
- *OpenVMS System Manager's Manual*
- *OpenVMS System Management Utilities Reference Manual: M-Z*
- *RMS Journaling for OpenVMS Manual*

For additional information on OpenVMS products and services, access the Compaq website at the following location:

<http://www.openvms.compaq.com>

## Reader's Comments

Compaq welcomes your comments on this manual. Please send comments to either of the following addresses

Internet	<b><a href="mailto:openvmsdoc@compaq.com">openvmsdoc@compaq.com</a></b>
Mail	Compaq Computer Corporation OSSG Documentation Group, ZKO3-4/U08 110 Spit Brook Rd. Nashua, NH 03062-2698

## How To Order Additional Documentation

Use the following World Wide Web address to order additional documentation:

<http://www.openvms.compaq.com/>

If you need help deciding which documentation best meets your needs, call 800-282-6672.

## Conventions

The following conventions are used in this manual:

Ctrl/ <i>x</i>	A sequence such as Ctrl/ <i>x</i> indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.
PF1 <i>x</i>	A sequence such as PF1 <i>x</i> indicates that you must first press and release the key labeled PF1 and then press and release another key or a pointing device button.
<span style="border: 1px solid black; padding: 2px;">Return</span>	In examples, a key name enclosed in a box indicates that you press a key on the keyboard. (In text, a key name is not enclosed in a box.)  In the HTML version of this document, this convention appears as brackets, rather than a box.
...	A horizontal ellipsis in examples indicates one of the following possibilities: <ul style="list-style-type: none"><li>• Additional optional arguments in a statement have been omitted.</li><li>• The preceding item or items can be repeated one or more times.</li><li>• Additional parameters, values, or other information can be entered.</li></ul>
. . . . . .	A vertical ellipsis indicates the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.
( )	In command format descriptions, parentheses indicate that you must enclose the options in parentheses if you choose more than one.
[ ]	In command format descriptions, brackets indicate optional elements. You can choose one, none, or all of the options. (Brackets are not optional, however, in the syntax of a directory name in an OpenVMS file specification or in the syntax of a substring specification in an assignment statement.)
[   ]	In command format descriptions, vertical bars separating items inside brackets indicate that you choose one, none, or more than one of the options.
{ }	In command format descriptions, braces indicate required elements; you must choose one of the options listed.
<b>bold text</b>	This text style represents the introduction of a new term or the name of an argument, an attribute, or a reason.
<i>italic text</i>	Italic text indicates important information, complete titles of manuals, or variables. Variables include information that varies in system output (Internal error <i>number</i> ), in command lines (/PRÓDUCER= <i>name</i> ), and in command parameters in text (where <i>dd</i> represents the predefined code for the device type).
UPPERCASE TEXT	Uppercase text indicates a command, the name of a routine, the name of a file, or the abbreviation for a system privilege.

Monospace text

Monospace type indicates code examples and interactive screen displays.

In the C programming language, monospace type in text identifies the following elements: keywords, the names of independently compiled external functions and files, syntax summaries, and references to variables or identifiers introduced in an example.

-

A hyphen at the end of a command format description, command line, or code line indicates that the command or statement continues on the following line.

numbers

All numbers in text are assumed to be decimal unless otherwise noted. Nondecimal radices—binary, octal, or hexadecimal—are explicitly indicated.



---

## DECram Disks

This chapter describes the DECram disk, which is a disk device created in the physical memory area of a system. The operating system can read from and write to a DECram disk, using standard OpenVMS disk I/O operations, at access times much greater than those for standard hardware disks. DECram disks use the DECram for OpenVMS device driver (MDDRIVER).

### 1.1 Characteristics of a DECram Disk

The DECram driver allows you to create a disk in available physical memory and to read and write to that disk using standard OpenVMS disk I/O operations. This provides high-speed access to read-only data such as libraries, fonts, and command files. Additionally, the DECram disk can be used to hold temporary or scratch files that an application may require.

Other characteristics include the following:

- A DECram disk can be accessed through the OpenVMS file system in the same way physical disks are accessed. It requires no change to the application or to the system software.
- Using a DECram disk reduces I/O traffic by replacing disk I/O with main memory access. While the number of I/O operations does not change, the number of external disk read and write operations is reduced.
- A DECram disk can be MSCP or QIO served in a OpenVMS Cluster configuration. The MSCP server implements mass storage control protocol software to make DECram disks accessible to all cluster members. For more information on served disks, refer to *OpenVMS Cluster Systems*.
- A DECram disk can be a member of a host-based shadow set. The shadow set can be made up of any combination of physical or DECram disks. Volume Shadowing for OpenVMS provides high data availability by duplicating data on multiple disks. If one disk fails, the remaining disk or disks can continue to service application and user I/O requests. See the *Volume Shadowing for OpenVMS* document for more information.

---

#### Caution

---

Because DECram disk data is resident in main memory, the data will be lost if the host system fails or is shut down. Therefore, Compaq recommends using the DECram disk to store small, frequently accessed files such as temporary (scratch) files, or read-only files such as commonly used image files that reside permanently on a conventional disk unless the DECram disk is shadowed to a physical disk with OpenVMS host-based volume shadowing.

---

## DECram Disks

### 1.1 Characteristics of a DECram Disk

#### 1.1.1 Comparing a DECram Disk to a Conventional Disk

Table 1–1 compares the features of a DECram disk to conventional disk features.

**Table 1–1 DECram Disk Versus Conventional Disk**

Type of Storage Device	
A virtual disk that operates like a physical disk device.	Disk device.
Device Setup	
Set up once each time the system is rebooted.	Set up at installation time. There is rarely, if ever, a need to set up the device again.
Models Available	
Only one model, although the size can vary. Upgrading is not a consideration.	Many models, each fixed in size. Upgrading is always an important consideration. Options include price/performance, size of the disk, density, and seek time.
Use as a Storage Device	
Typically holds small files. Expansibility is limited by the amount of system memory; cannot be used for offline storage.	Holds files of all sizes. Expansible to several orders of magnitude more than the amount of data that can be stored on a DECram disk. Some models have removable packs for secure offline storage.
Write Operations	
Stored data can be lost if system fails or is shut down. Writable, permanent files should generally not be placed on the DECram disk. However, if writable files must be placed on a DECram disk, then strongly consider the use of volume shadowing and journaling for data preservation.	Suitable for writable, permanent file storage.
Data Access Performance	
Extremely fast, limited only by CPU power and memory bandwidth, which are electronic in nature.	Speed limited by mechanical considerations, that is, seek time and rotation speed. Performance is also limited by other factors, such as interconnect bandwidth and controller features.

#### 1.1.2 Comparing a DECram Disk to Disk Caches

Both DECram disks and disk caches improve system performance by providing faster access to data. However, they differ in how they function and in how they are used by the system. Table 1–2 describes these differences.

## DECram Disks

### 1.1 Characteristics of a DECram Disk

**Table 1–2 DECram Disk Versus Disk Caches**

<b>Type of Storage Device</b>	
Operates like a disk device.	Operates as memory.
<b>Device Setup</b>	
Requires preloading files and using logical names to access these files on the DECram disk.	A disk cache is easily configured. The cache is usually transparent to the user.
<b>Use as a Storage Device</b>	
Typically holds entire files. Can hold as much as 4,294,967,296 blocks of information on one disk with Version 3.0. The maximum amount of information on one disk that Version 2.3 can hold is 524,280 blocks for a VAX system and 67,108,864 blocks on an Alpha system. It is possible to create logically contiguous devices greater than either of these limits by creating multiple DECram devices and binding them together.	Typically holds only portions of files. Capable of holding entire databases.
Both data blocks and file system blocks can be held. Consequently, opening and closing files is faster.	Disk caches often migrate to holding data blocks, as opposed to file system blocks.
<b>Type of Data</b>	
Choice of files to be held on the disk; subject to size constraints.	Choice of files to be held on disk caches not usually allowed.
<b>Write Operations</b>	
Not intended for permanent data storage. No write-back of data to permanent storage media.	Write operations write-back data to permanent storage media.
Generally more efficient than writing to a disk cache; scratch files are created relatively quickly.	Writing to a disk cache is slower than writing to a DECram disk. Creating scratch files is a relatively slow process.
Local DECram disk files cannot survive a system shutdown. Shared DECram disk files can survive a system shut down.	Data blocks survive a system shutdown.

(continued on next page)

## DECram Disks

### 1.1 Characteristics of a DECram Disk

**Table 1–2 (Cont.) DECram Disk Versus Disk Caches**

<b>Data Access Performance</b>	
Always reliable. Files are under the user's control and readily accessible.	Not always predictable. Access can vary by disk size and system load. (However, access is always more efficient than on comparable systems without any disk cache.)
<b>Recommendations for Use</b>	
Used for small, frequently accessed files such as application scratch files, system images, libraries, and DCL procedures.	Disk caches should be used for databases, general user work files, and files that increase in size.  Some disk caches use a protocol to write-back data where the software is informed that a write operation is complete before ensuring the data is actually stored on the disk. Although write-back by a disk cache increases write performance to a level comparable to that of a write to a DECram disk, data can be lost if the system shuts down before the modified data is written back to disk. Users should be aware that the write-back characteristics of a disk cache can result in lost data.

---

## DECram Disk Configuration

DECram Version 3.0 is designed specifically to take advantage of some of the advanced capabilities, such as clustering and Galaxy shared memory, of the newest Alpha systems. For example, one of the major differences between Version 2.3 and Version 3.0 capabilities is that Version 3.0 moves the virtual device addressing from the S1 address into the S2 address space. This change allows for the creation and addressing of devices larger than 2.0 gigabytes (GBs). Because Version 3.0 was engineered with many new features available on AlphaServer systems, the VAX/VMS system is not able to fully utilize all of the new features found in Version 3.0. Therefore, VAX/VMS systems will continue to be supported only on Version 2.3 of DECram because they cannot use those new features. DECram V3.0 is fully compatible with DECram V2.3. There can be any combination of these two versions of DECram in a VMScluster.

Any Alpha-based system can be easily upgraded to DECram Version 3.0. With Version 3.0 you can use the new DECram command interface or continue using the same familiar commands from SYSMAN for creating, initializing, and mounting DECram disks.

In DECram Version 3.0, you will find that a disk is configured using the DECram> user interface to create and format the Random Access Memory (RAM) disk. The INITIALIZE command is then used to write the OpenVMS cluster file system to the RAM disk.

The following sections describe how these utilities and commands are used to configure a DECram disk. Where applicable, and for the sake of clarity, reference is made to both versions for issuing the commands.

### 2.1 Preparing to Configure a DECram Disk

Before creating a DECram disk, the system manager must determine the size of the DECram disk to be created. Each disk block allocated to a DECram device uses 516 bytes of memory. See Section 2.2.2 for information on how to allocate space. In addition, the system manager should assess the need for multiple DECram units based on application and user demands.

The system manager must allocate main memory from the system for the DECram disk. Therefore, a substantial amount of additional system memory may be required, depending on the size of files to be placed on the RAM disk. The amount of memory dedicated to a DECram disk is determined by the amount of memory required for a particular application and the system resources required for the DECram disk. Section 2.1.1 describes the system resources required for DECram disks.

DECram Version 3.0 disks are created and formatted by the DECram> prompt and are initialized using the DCL INITIALIZE (INIT) command. If you are configuring DECram Version 3.0 or later on OpenVMS Alpha (Version 7.2-1H1 and later), you can generate a DECram startup procedure to set up the disk and

## DECram Disk Configuration

### 2.1 Preparing to Configure a DECram Disk

copy any required files to it. Usually, this procedure is called from the system startup procedure SYSSMANAGER:SYSTARTUP\_VMS.COM.

It is important to remember to check for disk errors after issuing any DECRAM command. Not all errors are returned to the user interface. Errors specific to a device are sent to the system errorlog. Type SHOW DEVICE MD at the DCL prompt to see if there were any device errors generated as a result of DECRAM command. You will need to use an errorlog analyzer tool to recover the error. However, the errors are logged in ASCII file format so you could search for errors with an MD-E-FAILURE prefix in the SYSSSYSROOT:[SYSERR]ERRLOG.SYS file.

#### 2.1.1 Resource Requirements

The DECRAM for OpenVMS device driver is a set of routines and tables that the OpenVMS operating system uses to process an I/O request for a DECRAM disk device. In addition to the nonpaged pool requirements for the driver and the I/O database structures (see the *OpenVMS VAX Device Support Manual* available on the OpenVMS Documentation CD-ROM), the DECRAM disk uses system memory, taken from the free page list, as device storage. You can specify the amount of memory required during the configuration procedure.

On OpenVMS Alpha systems (Version 7.2-1H1 or later) running DECRAM Version 3.0, the only requirements are that a DECRAM disk must have 516 bytes of free page list per block (512 bytes) of disk space allocated.

## 2.2 Configuring DECRAM Disks

Under DECRAM Version 3.0, you can configure a local DECRAM disk using either the command procedures for DECRAM Version 2.3 or the new command procedures for Version 3.0 using the DECRAM> interface.

To configure a DECRAM disk under Version 3.0, you must perform the following tasks:

- Create a DECRAM disk (see Section 2.2.1)
- Initialize the disk (see Section 2.2.2.2)
- Mount the disk (see Section 2.2.3)

#### Shadowed DECRAM Disks

You can configure shadowed DECRAM disks on OpenVMS Cluster members running OpenVMS VAX Version 5.5 or later, or OpenVMS Alpha Version 6.1 or later.

The shadowing of a DECRAM disk to a real physical disk is strongly recommended when using OpenVMS Alpha Version 7.3 and Version 7.2-1H1. The reason for this recommendation is to avoid a loss of data during an incomplete write operation. DECRAM writes data a block at a time. If the system experiences a power loss or other interruption of service and the system is in the process of writing, the block write would not complete successfully and the data would be lost. There is no way to recover the block of information because DECRAM data is volatile. One way to protect against this condition is to shadow the DECRAM disk with a physical disk so that data can be saved to a non-volatile medium and be recovered.

See the *Volume Shadowing for OpenVMS* manual for complete information about creating and using host-based shadow sets.

### 2.2.1 Creating a DECram Disk

The method used to create a DECram disk depends on which version of DECram and the OpenVMS operating system you are running, as shown in Table 2-1.

**Table 2-1 Creating a DECram Disk**

Platform	OpenVMS Version	Type of Disk Support	Section to Read
<b>DECram Version 3.0 or later</b>			
VAX	All versions	Version 3.0 not supported	Not Applicable
Alpha	Version 7.21H1 and later Versions prior to Version 7.2-1H1	Local and shared Version 3.0 not supported	Section 2.2.1.2 Not Applicable

#### 2.2.1.1 OpenVMS Alpha All VMS Versions to Create Local Disks

For Alpha systems running a version of OpenVMS prior to Version 7.2-1H1, you must continue to use the System Management Utility SYSMAN. However, if you upgrade to DECram Version 3.0, then you may either continue to use the SYSMAN utility or use the DECGRAM> command interface.

For example: Under the DECram Version 3.0 software, you can use the DECGRAM> interface by issuing the following command:

```
$ DECGRAM CREATE DISK ddcuuuuu: /CAPACITY=blocks /MEMORY=type [/qualifier]
```

Or continue to use the SYSMAN command:

```
SYSMAN> IO CONNECT ddcuuuuu:/DRIVER=SYS$MDDRIVER/NOADAPTER
```

where:

**ddcuuuuu:** represents the device name.

**dd**

The device code is always MD for a disk device.

**c**

The controller designator can be any letter, A through Z.

**uuuuu**

The unit number can be any value from 0 to 32767.

**Example:**

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> IO CONNECT MDA0:/DRIVER=SYS$MDDRIVER/NOADAPTER
SYSMAN> EXIT
```

The command in this example creates device MDA0 and connects it to the DECram driver (MDDRIVER).

See the *OpenVMS System Management Utilities Reference Manual* for more information on the SYSMAN command IO CONNECT.

## DECram Disk Configuration

### 2.2 Configuring DECram Disks

#### 2.2.1.2 OpenVMS Alpha Version 7.2-1H1 and Later: Using DECram Version 3.0 (or Later) to Create Local and Shared Disks

DECram Version 3.0 allows you to create disks that can be shared by other nodes in the cluster. Detailed information for creating local and shared disks using the CREATE command and associated qualifiers can be found in Section 2.2.2.2.

#### 2.2.2 Allocating and Deallocating Space for a DECram Disk

Under DECram Version 3.0, there is no longer a limit on system space per block of disk space allocated.

A Galaxy configuration is required for creating a DECram shared disk. A stand-alone system in a Galaxy can access the shared disk but a Galaxy and a VMScluster is required to access a shared disk from multiple instances in the Galaxy or multiple nodes in the VMScluster. Therefore, shared disks are supported only on OpenVMS Alpha Version 7.2-1H1 (or later) on systems that are capable of supporting Galaxy shared memory.

For more information on local or shared disks for DECram Version 3.0 (or later), see Section 2.2.2.2.

#### 2.2.2.1 Allocating and Deallocating Space Using INITIALIZE

You can allocate or deallocate memory space for a local DECram disk by using the DCL command INITIALIZE:

```
$ INITIALIZE ddcuuuuu: volume-label /SIZE=n
```

where:

**ddcuuuuu:**

The device name used in the SYSGEN command CONNECT to create the device (see Section 2.2.1.1).

**volume-label**

A unique disk label up to 12 characters maximum.

**/SIZE=n**

The amount of memory (in blocks or 512 byte chunks) to be allocated to this device.

---

**Restriction**

---

With DECram for OpenVMS Version 3.0, the limit on the DECram disk size has been extended to 4,294,967,296 blocks on OpenVMS Alpha systems running Version 7.2-1H1 or higher. Under DECram Version 2.3, the maximum disk size on an OpenVMS Alpha system is 67,108,864 blocks and on an OpenVMS VAX system, disk size is limited to 524,280 blocks. It is possible to create logically contiguous devices greater than either of these limits by creating multiple DECram devices and binding them together.

---



## DECram Disk Configuration 2.2 Configuring DECram Disks

### Deallocating Disk Space

To **deallocate** memory used by a DECram disk, specify `/SIZE=0` in a subsequent `INITIALIZE` command. This returns all memory resources to the system.

---

#### Warning

---

If you initialize a DECram disk, all data on the disk is lost.

---

Refer to the online help available through DECram or see the *OpenVMS DCL Dictionary* for more information on the `INITIALIZE` command.

#### Example:

```
$ INITIALIZE/INDEX-BEGINNING MDA0 FASTRAMDISK /SIZE=300
```

The command in this example initializes the device `MDA0` with a size of 300 blocks. The `/INDEX-BEGINNING` qualifier places the index file for the volume's directory structure at the beginning of the volume. The volume label is `FASTRAMDISK`. The following command deallocates the same disk by setting `/SIZE` to 0.

```
$ INITIALIZE MDA0 FASTRAMDISK /SIZE=0
```

### INITIALIZE Error Messages

The following error messages can be returned for the DCL command `INITIALIZE`:

- `%INIT-F-DUPUNIT`, duplicate unit number detected by MSCP controller  
The name of the device being initialized is not unique; that is, another device with this name exists in the cluster. Specify a unique name for the device to initialize it successfully.
- `%INIT-F-INSFMEM`, insufficient dynamic memory  
There is insufficient memory to initialize a disk of the size specified. Specify a smaller disk or add more memory.

### Initialization Failure and Recovery

If disk initialization fails due to insufficient resources, you can attempt recovery by performing one or more of the following steps:

- Acquire more space by deallocating other DECram disks that are no longer required. (Specify `/SIZE=0` in the `INITIALIZE` command.)
- Reduce the value of `/SIZE` to create a smaller DECram disk.
- Increase main memory or use a node with greater memory.

#### 2.2.2.2 Allocating and Deallocating Space Using DECram Version 3.0 and Later

With DECram Version 3.0 and later, you can allocate space for a local DECram disk or, on OpenVMS Alpha Version 7.2-1H1 (or later), for a disk in Galaxy shared memory. Local memory is accessible only from the local node or instance. Shared memory is accessible from any Galaxy instance that attaches to the shared memory section.

To allocate space, you must have the following privileges: `AUDIT`, `CMKRNL`, `SYSLCK`, `SYSPRV`, and `PHY_IO`.

Enter the following command from the DCL prompt:

```
$ DECram CREATE DISK device-name /CAPACITY=blocks /MEMORY=type [/qualifier]
```

where:

## DECram Disk Configuration

### 2.2 Configuring DECram Disks

#### **device-name**

The name of the virtual memory disk that you want to create or modify in local memory or in Galaxy shared memory.

The disk name takes the form **ddcuuuuu**, where **dd** is always MD, the controller **c** can be a letter from A through Z, and the disk unit number **uuuuu** can be any number in the range 0 through 32767 (for example, MDC256).

#### **/CAPACITY=blocks**

This required qualifier specifies the size in blocks (512 bytes=1 block) of the virtual disk to be created or modified. The capacity block limit in DECram Version 3.0 is now 8,388,608 blocks.

#### **/MEMORY=type**

This is a required qualifier with no default. You must specify the memory type as either LOCAL or SHARED. The type of memory created will depend on the type of data structures needed to support the applications and users on that system. Once you have created a specific memory type, there is overhead memory that is not deallocated when the DECram disk size is set to zero.

#### **/[NO]ALLOCLASS=class-number**

This optional qualifier specifies an allocation class to be permanently associated with the controller. This qualifier is allowed for SHARED memory type disks only. The default allocation class value is 99.

---

#### **Note**

---

Once you use /ALLOCLASS to associate a controller letter with a specific allocation class, this association remains in effect as long as the shared memory section and the MDRECOVER.DAT file both exist.

---

#### **/[NO]PERSIST**

This optional qualifier controls whether or not the disk is restored after a reboot. The default qualifier, if not stated, is /PERSIST.

Following are the three ways to restore a disk:

- Execute the SYSMAN command MDRECOVER image at startup time (see Section 2.3.1).
- Execute the DCL command DECRAM>RECOVER (see Section 2.3.2).
- Execute the DECRAM>CREATE DISK command again (see Section 2.2).

#### **/[NO]SERVE**

This optional qualifier controls whether or not disk serving is enabled for the OpenVMS Cluster system. The default qualifier, if not stated, is /NOSERVE.

Once a disk has been specified as /SERVE, it cannot be selected as /NOSERVE. If you create disks on one node with the /NOSERVE qualifier and then try to mount that disk cluster wide with the MOUNT/CLUSTER command then the process where the MOUNT/CLUSTER command was issued will hang. A DECram disk can be changed from NOSERVE to SERVE status at any time by issuing the original DECram CREATE DISK command with the SERVE qualifier. If you intend to do a cluster-wide mount then you must create the DECram disk using the /SERVE qualifier.

## DECram Disk Configuration 2.2 Configuring DECram Disks

A DECram disk can be changed to be served to the cluster at any time by issuing the same DECram CREATE DISK command with the /SERVE qualifier. It is not necessary to delete the DECram disk first. This is the only attribute that can be modified while the disk size is greater than zero.

### **/[NO]WRITEBUFFERED**

When specified, this optional parameter helps reduce write recovery time after a power failure or system crash (default is NOWRITEBUFFERED). You can also employ other application recovery techniques and OpenVMS Volume Shadowing to reduce write recovery time after a power failure.

---

#### **Note**

---

While this parameter provides for higher levels of availability, it significantly slows down I/O to the DECram disks. This parameter is **NOT** currently implemented in DECram Version 3.0. It is included to make you aware that this option will be available and can be used with volume shadowing to reduce the write recovery time after an interruption of service. See Section 2.2 for more information related to this topic.

---

### **Examples:**

```
$ DECram CREATE DISK MDB150 /CAPACITY=600 /MEMORY=SHARED /ALLOCLASS=5
```

The command in this example creates the shared disk MDB150 with a size of 600 blocks. The shared disk controller is permanently associated with allocation class 5.

The following example deallocates the same disk by setting the /CAPACITY qualifier to 0.

```
$ DECram CREATE DISK MDB150 /CAPACITY=0 /MEMORY=SHARED /ALLOCLASS=5
```

---

#### **Warning**

---

When deallocating, you must specify the command line qualifiers in exactly the same way shown in the example above when creating the DECram disk except with the qualifier /CAPACITY=0. If you do not do this, you will get an unsupported error message and the disk will NOT be deallocated.

---

In general an unsupported error message, %SYSTEM-E-UNSUPPORTED, unsupported operation or function, means that you have entered an invalid command and/or command qualifier. You should review commands and qualifiers in DECram help and then re-enter the new command and/or qualifier.

The following example initializes a RAM disk:

```
$ INIT MDB150 FASTRAMDISK
```

## DECram Disk Configuration

### 2.2 Configuring DECram Disks

#### 2.2.2.2.1 Special Notes for Shared Disks During Allocation and Deallocation

---

##### Note

---

Shared disks only need to be initialized once, the first time they are created.

---

When a DECram shared disk is created on one instance it automatically appears on all other instances in the Galaxy if at least one shared disk has been created on each of the instances in the Galaxy. For example, if DECram was installed on each instance in the Galaxy with the shared disk option during installation then one shared disk would have been created on each node.

You cannot change the size of a shared disk while other nodes are attached to it. You must set the size of the shared disk to zero on all nodes before you can change the size of a shared disk.

DECram recovers shared disks after a power failure and attaches them to existing shared memory regions. This instantly restores DECram data that was available prior to the crash. The recovery data is located in a file called MDRECOVER.DAT and works best if this file is located on a common VMS system disk. If each node has its own MDRECOVER.DAT file then it might not restore the same disks as other Galaxy instances. If a common system disk is not possible then it is up to the system manager to keep the multiple copies of the MDRECOVER.DAT file consistent by copying the master to all instances at SYSS\$COMMONH:[SYSMGR]MDRECOVER.DAT.

The DECram version 3.0 driver does not support VMS FAST IO, (i.e., the IOS\_PERFORM QIO function).

### 2.2.3 Mounting a DECram Disk

The method for mounting a DECram disk varies, depending on whether the disk is local or shared, as follows:

- Mounting a local DECram disk, see Section 2.2.4
- Mounting a shared DECram disk, see Section 2.2.5

### 2.2.4 Mounting a Local DECram Disk

You can mount a local DECram disk using the following syntax of the DCL command MOUNT:

```
$ MOUNT/NOCACHE device-name: volume-label
```

#### **device-name**

The device name used to create the device in the SYSMAN command IO CONNECT on Alpha systems or at the DECram> prompt by issuing the CREATE/MEM=LOCAL qualifiers (see Section 2.2.1).

#### **volume-label**

The unique disk label used in the INITIALIZE command.

### Example

```
$ MOUNT/NOCACHE MDA0: FASTERDISK
```

The command in this example mounts the device MDA0. The /NOCACHE qualifier disables caching limits established at system generation time. The volume label is FASTERDISK.

---

### Note

---

Data on the DECram disk is preserved if you dismount and then mount the disk.

---

See the *OpenVMS System Management Utilities Reference Manual* for more information on the MOUNT command.

### 2.2.5 Mounting a Shared DECram Disk

Use the following syntax of the DCL command MOUNT to mount a shared disk for single or clustered systems:

- For a single standalone system:

```
$ MOUNT/SYSTEM/NOCACHE MDB150 FASTERDISK
```

---

### Warning

---

The default for a normal hard disk when using the MOUNT command is to the CACHE qualifier. Therefore, you need to specify the /NOCACHE qualifier when using a RAM disk. There is no benefit to using the CACHE qualifier, and if used, would actually waste cache memory resources.

---

- For a clustered system:

```
$ MOUNT/CLUSTER/NOCACHE MDB150 FASTERDISK
```

The DECram disk then can be accessed by any instance that is attached to the shared disk region.

### Example

```
$ MOUNT/NOCACHE MDB150 VOLUME-LABEL
```

The command in this example mounts the shared device MDB150. If you want to share the RAM disk, then you must specify the /SYSTEM or /CLUSTER qualifiers.

---

### Note

---

Data on the shared DECram disk is preserved if you dismount and then mount the disk. In addition, data is also preserved on the DECram disk even if the Galaxy instance crashes, as long as another instance is attached to the shared region.

---

See the *OpenVMS System Management Utilities Reference Manual* for more information on the MOUNT command.

## DECram Disk Configuration

### 2.3 Restoring an OpenVMS Alpha Version 7.2-1H1 (or Later) DECram Disk

#### 2.3 Restoring an OpenVMS Alpha Version 7.2-1H1 (or Later) DECram Disk

Following are the three ways to restore all RAM disks:

- Execute the SYSMAN command MDRECOVER at startup time (see Section 2.3.1).
- Execute the DCL command DECRAM RECOVER (see Section 2.3.2).
- Execute the DECRAM>CREATE DISK command again (see Section 2.2)

##### 2.3.1 Restoring a Disk Using SYSMAN STARTUP

The MDRECOVER.EXE image restores memory-based disks after a power failure or system crash. MDRECOVER.EXE uses memory disk data in SYS\$COMMON:[SYSMGR]MDRECOVER.DAT to recover memory disks with the /PERSIST qualifier set.

---

##### Note

---

Separate MDRECOVER.DAT files may not have the exact same list of disks in them and that a recovery from one node may not restore the disks of another cluster node. A common MDRECOVER.DAT file is required for consistent cluster-wide memory disk recovery. This file is invoked automatically when the system is rebooted.

---

Issuing the "PRODUCT INSTALL DECRAM" statement at the DCL prompt invokes the POLYCENTER Software Installation procedure. POLYCENTER then installs the following command in the MDRECOVER.DAT file for execution by OpenVMS at system startup time:

```
$ MCR SYSMAN STARTUP ADD FILE MDRECOVER.EXE/PHASE=LPMAIN/MODE=DIRECT
```

This command is issued automatically during installation and results in the restoration of the RAM disk(s) automatically after a power failure, system maintenance, or interruption of service for any reason.

##### 2.3.2 Restoring a Disk Using the DCL command DECRAM RECOVER

The following command can be issued to recover a DECram disk:

```
$ DECRAM RECOVER
```

The DECRAM command RECOVER reads from the MDRECOVER.DAT file located in the SYS\$COMMON:[SYSMGR] area and restores each disk in the file that has the /PERSIST qualifier set.

The following DECRAM command recovers all the disks with /PERSIST set:

```
$ DECRAM RECOVER
```

## 2.4 Determining Allocation of DECram Device Resources

The SHOW DEVICE command is used to show the resources that are allocated for existing RAM disks.

The DECRAM SHOW DISK command shows all of the disk resources that will be allocated on the next disk recovery when using the SYSSCOMMON:[SYSMGR]MDRECOVER.DAT file. That command includes all disks with or without the qualifier /PERSIST being set.

---

**Note**

---

If MDRECOVER.DAT is a cluster wide common file, it will not list local disks of other cluster nodes with different allocation classes.

---

### 2.4.1 Determining Resource Allocation Using DECram Version 3.0 and Later

When using DECram Version 3.0 and later, you can display information about a virtual disk from both the DECRAMSRECOVER.DAT and MDRECOVER.DAT recovery files by issuing the DCL command DECRAM SHOW DISK.

```
$ DECRAM SHOW DISK [device-name]
```

#### **device-name**

The device name used in the CREATE DISK command (see Section 2.2.2.2). If no device name is specified, a brief status is displayed for all disks in the recovery files.

When the command includes a disk specification, the following information is displayed about the disk:

- Disk name (as specified in the CREATE DISK command)
- Size in blocks
- Current volume label (if available)
- Type of memory: Local or Shared
- The persist, serve, and writebuffered status

When no device name is specified, only the disk name, size, and volume label (if available) is displayed for all disks in the recovery files.

If no output is displayed, no DECRAMSRECOVER.DAT or MDRECOVER.DAT files exist.

Reference the online help information available through the OpenVMS DCL HELP command or see the *OpenVMS DCL Dictionary* for more information on the SHOW DISK command.

#### **Example**

```
$ DECRAM SHOW DISK
```

This command does not specify a device, so a brief summary of information is displayed for all devices in the DECRAMSRECOVER.DAT and MDRECOVER.DAT recovery files.

## DECram Disk Configuration

### 2.4 Determining Allocation of DECram Device Resources

```
From SYS$COMMON:[SYSMGR]MDRECOVER.DAT
Disk $4$MDA0 Size 110 Label MDA0
Disk $4$MDF0 Size 100 Label MDF0MDF0
Disk $4$MDF1 Size 100 Label MDF1MDF1
Disk $4$MDF2 Size 100 Label MDF2MDF2
Disk $99$MDE0 Size 100 Label MDE0
```

```
$ SHOW DISK MDA0
```

```
From SYS$COMMON:[SYSMGR]MDRECOVER.DAT
Disk $4$MDA0 Size 110 Label MDA0
Local memory, persist, serve, writebuffered.
```

This command displays full information about device MDA0.

---

#### Warning

---

Never delete the MDRECOVER.DAT file. If you delete the file, then you will not be able to recover any disks after a power failure.

---

## 2.5 Shadowing a DECram Disk

The algorithm used by shadowing to select shadow set members for an IO operation is based primarily on member queue lengths. Although the queue lengths of all shadow set members will almost always be the same, DECram is expected to do more IO since it returns reads and writes much more quickly than a physical disk. One way to ensure that the application reads only from shadow set DECram disks is to do IO local to the DECram disk while the other shadow set members are served.

---

#### Warning

---

If you use the INIT/SHADOW DCL command to initialize a DECram disk followed by a MOUNT/SHADOW command, be aware that OpenVMS volume shadowing will not execute a full copy operation. To execute a full copy operation of shadowed disks and make the data fully consistent between shadow set members, you should issue an INIT/ERASE command after the INIT/SHADOW command.

---



### 2.5.1 Mounting a Shadowed DECram Disk–DECram Disk to DECram Disk

To mount a shadowed DECram disk, use the following format with the DCL command MOUNT for single or clustered systems:

```
$ MOUNT DScuuuu/shadow=( $n$ $device-name:) volume-label
```

- For a single standalone system:

```
$ MOUNT/SYSTEM/NOCACHE DSA0: /SHADOW=( $3$ $MDB100,  $3$ $MDA0) FASTRAMDISK
```

- For a clustered system:

```
$ MOUNT/CLUSTER/NOCACHE DSA0: /SHADOW=( $3$ $MDB100,  $2$ $MDA0) FASTRAMDISK
```

where:

**DS**

Represents the controller code identifier, which is always DS for a shadowed disk.

**c**

Represents any single letter, A to Z.

**uuuuu**

Represents any numeric value between 0 and 32,767.

**n**

Represents the allocation class for this OpenVMS system.

**device-name**

The full device-name returned by SHOW DEVICE *device-name* (see Section 2.2.1).

**volume-label**

The unique disk label used in the INITIALIZE command.

**Example**

```
$ MOUNT DSA0/NOCACHE/SHADOW=( $2$ $MDA1000:) FASTRAMDISK
```

---

**Reminder**

Never issue a MOUNT command without the /NOCACHE qualifier.

---

The command in this example mounts the host-based shadow set DSA0 with the single member  $2$ \$MDA1000.

The volume label is FASTRAMDISK.

---

**Note**

Data on the shadowed DECram disk is preserved if you dismount and then mount the host-based shadow set.

---

See the manual *Volume Shadowing for OpenVMS* for more information on allocation class and the MOUNT command. *OpenVMS System Management Utilities Reference Manual: M–Z* contains additional information on the MOUNT command.

## DECram Disk Configuration

### 2.5 Shadowing a DECram Disk

#### 2.5.2 Mounting a Shadowed DECram Disk—DECram Disk to Physical Disk or Partition

DECram can be used to shadow real physical devices or partitions provided there is enough physical memory available. You would do this in any situation where, in addition to data integrity, application speed and performance are also important.

For reads, the shadowing code will read from the DECram disk. Writes will be written to all devices and, therefore, will be much slower than a write to a DECram disk only. On power failure, the data will be safe because it is stored on physical media.

---

#### Warning

---

Volume Shadowing for OpenVMS does not currently freeze writes to the shadow set when the physical disk is no longer available. Therefore, be aware that in the current Version 3.0 release, if the physical disk fails, you will be writing information to a volatile disk and data may be permanently lost if the write is interrupted due to a power failure.

---

If you have an existing shadow set that you know has more reads than writes, placing a DECram disk in the shadow set can dramatically improve overall performance. The reason for this performance improvement is that a read operation does not require accessing a physical disk because the information is available on the RAM disk. Therefore, the time required to access the data is minimal. A write operation in a shadow set comprised of a DECram disk still requires the time needed to write the data out to the physical disk.

##### 2.5.2.1 Creating the DECram Disk

Use the following command to determine the total number of disk blocks your physical device (or existing DSA device) has; this is the size you must use to create your DECram disk:

```
$ SHOW DEVICE/FULL $n$device-name:
```

Use the DECram CREATE command to create your DECram disk with the /CAPACITY qualifier equal to total blocks. Then use the MOUNT command to add the DECram disk to the shadow set and initiate the shadow copy to the DECram disk. These commands are shown in the following example.

```
$ show dev dsa640/full
```

```
Disk DSA640:,device type MSCP served SCSI disk, is online, mounted, file-
oriented device, shareable, available to cluster, error logging is enabled.
```

Error count	0	Operations completed	8107
Owner process	"	Owner UIC	[SYSTEM]
Owner process ID	00000000	Dev Prot	S:RWPL,O:RWPL,G:R,W
Reference count	1	Default buffer size	512
** Total blocks	1027362	Sectors per track	85
Total cylinders	756	Tracks per cylinder	16

## DECram Disk Configuration 2.5 Shadowing a DECram Disk

```

Volume label           "TST640"      Relative volume number      0
Cluster size          3              Transaction count          1
Free blocks           1027203      Maximum files allowed      128420
Extend quantity       5              Mount count                2
Mount status          System        Cache name                 "_$84$DKC200:XQPCACHE"
Extent cache size     64            Maximum blocks in extent cache 102720
File ID cache size    64            Blocks currently in extent cache 0
Quota cache size      0             Maximum buffers in FCP cache 4610
Volume owner UIC      [SYSTEM]      Vol Prot                   S:RWCD,O:RWCD,G:RWCD,W:RWCD

```

Volume Status: subject to mount verification, file high-water marking, write-back caching enabled.  
Volume is also mounted on CSG6.

Disk \$4\$DUA640:, device type MSCP served SCSI disk, is online, member of shadow set DSA640:, error logging is enabled.

```

Error count           0              Shadow member operation count 8126
Host name             "HSJ50R"      Host type, avail           HSJ5, yes
Alternate host name   "HSJ50G"      Alt. type, avail           HSJ5, yes
Allocation class      4

```

Disk \$4\$DUA642:, device type MSCP served SCSI disk, is online, member of shadow set DSA640:, error logging is enabled.

```

Error count           0              Shadow member operation count 8134
Host name             "HSJ50R"      Host type, avail           HSJ5, yes
Alternate host name   "HSJ50G"      Alt. type, avail           HSJ5, yes
Allocation class      4

```

```

$ DECRAM CREATE DISK
MDA1642/CAP=1027362/MEMORY=LOCAL/SERVE/ALLOCLASS=84

$ mount/cluster/nocache dsa640/shad=$84$MDA1642: TST640
%MOUNT-I-MOUNTED, TST640 mounted on _DSA640:
%MOUNT-I-SHDWMEMCOPY, _$84$MDA1642: (CSG84) added to the shadow set with a copy
operation
%MOUNT-I-ISAMBR, _$4$DUA642: (HSJ50R) is a member of the shadow set
%MOUNT-I-ISAMBR, _$4$DUA640: (HSJ50R) is a member of the shadow set

$ show dev dsa640

Device      Device      Error  Volume  Free  Trans  Mnt
Name        Status      Count Label   Blocks Count  Cnt
DSA640:     Mounted    0      TST640 1027203 1      5
$4$DUA640: (HSJ50R) ShadowSetMember 0 (member of DSA640:)
$4$DUA642: (HSJ50R) ShadowSetMember 0 (member of DSA640:)
$84$MDA1642: (CSG84) ShadowCopying 0 (copy trgt DSA640: 1% copied)

$

```

### 2.5.2.2 Mounting a Shadowed DECram Disk

To mount a shadowed DECram disk, use the following format with the MOUNT command, using the existing shadow set device name:

```
$ MOUNT DScuuuu/shadow=( $n$device-name:) volume-label
```

where:

#### **DS**

Represents the controller code identifier, which is always DS for a shadowed disk.

#### **c**

Represents any single letter, A to Z.

#### **uuuuu**

Represents any numeric value between 0 and 32,767.

## DECram Disk Configuration

### 2.5 Shadowing a DECram Disk

**n**

Represents the allocation class for this OpenVMS Cluster member.

**device-name**

The full device-name returned by SHOW DEVICE *device-name* (see Section 2.2.1).

**volume-label**

The unique disk label used in the INITIALIZE command.

#### **Example**

```
$ MOUNT DSA0/NOCACHE/SHADOW=( $2$MDA1000: , $2$DUA546 ) FASTRAMDISK
```

The command in this example mounts the host-based shadow set DSA0 with the members \$2\$MDA1000 and \$2\$DUA546. The volume label is FASTRAMDISK. See Section 2.5.2.2 for more information on mounting a DECram disk.

See the manual *Volume Shadowing for OpenVMS* for more information on allocation class and the MOUNT command. *OpenVMS System Management Utilities Reference Manual: M-Z* contains additional information on the MOUNT command.

---

## Using a DECram Disk

This chapter describes applications that may benefit the most from using DECram disks, how to identify and locate files that can be stored on a DECram disk, and how using a DECram device might adversely affect system operations.

### 3.1 Recommended Applications for a DECram Disk

There are two general classes of applications that benefit substantially from using a DECram disk:

- Applications that frequently use system images
- Modular applications that produce temporary, transient files

Applications that frequently use system images execute a DCL command file in response to a user command. By placing the images and their associated libraries as well as heavily accessed data files on a DECram disk, execution speed improves. While it is possible to install some of the shared images as resident and thereby improve performance, the DECram disk can be used for data access, command procedures, or any other files that you simply want to run faster.

If the system fails or is shut down, you must reinitialize and then remount the local DECram disk. The system startup file must invoke a command procedure that can perform these functions. Shared disks are automatically created. A system startup file can mount the shared DECram disk for immediate use of saved data.

---

#### Note

---

If system files are installed on a DECram device, you cannot dismount that DECram device while system files are installed or open. Installed system files are open as long as they are mapped into a process.

---

Modular applications often accept input files, perform operations on the data, and write output files. These output files are then read by another program that produces other files, and so on. The application runs with greater efficiency if these intermediate input and output files are stored on a DECram disk. If the system fails or is shut down, it is necessary to restart the application from the beginning, or from a checkpoint if the application was a long-running continuous process.

## Using a DECram Disk

### 3.1 Recommended Applications for a DECram Disk

#### 3.1.1 Identifying Commonly Used Images

You can use the LIST command provided by the OpenVMS Install Utility (INSTALL) to identify commonly used images that may benefit from being stored on a DECram device. INSTALL tracks the number of times an image was accessed since installation.

By executing the INSTALL LIST command periodically and examining the output, you can determine the most frequently used images, as shown in the following example:

```
$ INSTALL LIST/FULL
```

This command produces output that is similar to what follows:

```
DISK$DUA0:<SYS0.SYSCOMMON.SYSEXE>.EXE
  ANALIMDMP;1                      Prv
    Entry access count              = 0
    Privileges = CMKRNL CMEXEC

  AUTHORIZE;1                      Prv
    Entry access count              = 0
    Privileges = CMKRNL

  CDU;1                            Open Hdr Prv
    Entry access count              = 0
    Current / Maximum shared        = 0 / 0
    Privileges = CMEXEC

  COPY;1                            Open Hdr Shar
    Entry access count              = 12
    Current / Maximum shared        = 0 / 2
    Global section count            = 1

  DCL;1                            Open Hdr Shar Lnkbl
    Entry access count              = 21
    Current / Maximum shared        = 0 / 3
    Global section count            = 1
```

The output shows that only the COPY.EXE and DCL.EXE files, as indicated by the entry access count, have been accessed since the last installation. This indicates that they might be potential candidates for storage on a DECram device.

#### 3.1.2 Using DECram Disks with a Workstation

Diskless workstations that access conventional disks over Ethernet can also use a DECram disk to improve performance. Because the most commonly used images are stored on the DECram disk, page faults and image activation occur locally, rather than over the Ethernet. While products such as XFC (a dynamic cluster caching product) can adjust over time to provide a high hit rate for frequently accessed images, only DECram can provide you with a 100 percent hit rate.

#### 3.1.3 Storing SYS\$SCRATCH Files on a DECram Disk

You can significantly improve system performance by using a DECram disk to hold scratch files produced by OpenVMS images, shareable images, and layered products.

Table 3-1 lists some of the OpenVMS images, shareable images, and layered products that open temporary files on SYSSCRATCH.

## 3.1 Recommended Applications for a DECram Disk

---

**Warning**

---

If the system on which the device physically exists either fails or is taken out of service, any journal or scratch files created by the images, shareable images, and layered products listed in Table 3–1 will be permanently lost.

---

**Table 3–1 Sources of Temporary Files**

SYSSLIBRARY:ADARTL.EXE	SYSSSYSTEM:DTM.EXE
SYSSLIBRARY:CMS\$MAILSHRP.EXE	SYSSSYSTEM:EDT.EXE
SYSSLIBRARY:CONVSHR.EXE	SYSSSYSTEM:MAIL.EXE
SYSSLIBRARY:EDTSHR.EXE	SYSSSYSTEM:NOTES.EXE
SYSSLIBRARY:MAILSHRP.EXE	SYSSSYSTEM:PCA.EXE
SYSSLIBRARY:NOTES\$MAILSHRP.EXE	SYSSSYSTEM:SORTMERGE.EXE
SYSSLIBRARY:PPLSHR.EXE	SYSSSYSTEM:VERIFY.EXE
SYSSLIBRARY:SMGSHR.EXE	SYSSSYSTEM:VMSTAILOR.EXE
SYSSLIBRARY:SORTSHR.EXE	

---

To access SYSS\$SCRATCH files that are sent to a DECram disk, each user must have a directory on the DECram disk and define the logical name SYSS\$SCRATCH to point to that directory.

You can create a directory by using the following format at the DCL command CREATE/DIRECTORY:

```
$ CREATE/DIRECTORY MDA0:[directory]
```

You must define the logical name SYSS\$SCRATCH to point to the directory on the DECram disk by using the following format:

```
$ DEFINE SYSS$SCRATCH MDA0:[directory]
```

where **MDA0** represents the name of the DECram disk, and **directory** represents the name of the user's directory on the DECram disk.

**3.1.4 Recovering Writable DECram Files**

Compaq recommends that you store only small, frequently accessed files, such as temporary (scratch) files or read-only files, because data can be permanently lost if the node on which it exists fails or if the DECram device is reinitialized.

If you have stored and lost writable data, you can recover those files by using OpenVMS RMS after-image journaling (AIJ). This is a layered product and requires a usage license. It can be used to recover critical, writable files stored on the DECram device. See the *RMS Journaling for OpenVMS Manual* (available on the OpenVMS documentation CD-ROM) for more information on RMS after-image journaling.

## Using a DECram Disk

### 3.2 Using a Search List to Locate Files

### 3.2 Using a Search List to Locate Files

You can use a search list to move a subset of files from a conventional disk to a DECram disk. A search list instructs the operating system to look for files on the DECram disk first, before looking for the files on a conventional disk. A search list is a logical name that has more than one translation, as shown in the following example:

```
$ DEFINE WORKFILE [WILSON.NOTES],[WILSON.WORK]
$ SHOW LOGICAL WORKFILE
  "WORKFILE" = "[WILSON.NOTES]" (LNM$PROCESS_TABLE)
              = "[WILSON.WORK]"
```

If you use the logical name `CONSTRUCTION` to point to the directory `SYS$SYSDEVICE:[CONSTRUCTION]`, and you have moved only a subset of the files in that directory to the DECram disk, redefine the logical name `CONSTRUCTION` as a search list, as shown in the following example:

```
$ DEFINE/SYSTEM CONSTRUCTION "MDA0:[CONSTRUCTION]", -
_.$ "SYS$SYSDEVICE:[CONSTRUCTION]"
```

Once the logical name has been redefined, any file lookup that uses the search list will examine each translation of the logical name to find the file. In this example, the `CONSTRUCTION` directory on the DECram disk will always be searched first. Defining a search list allows you to place a subset of the files addressed by a logical name on a DECram disk, while leaving the other files on the conventional disk. Any new file creations that use the search list will be created on the DECram disk because it is the first translation of the search list.

---

#### Note

---

If you redefine system logical names and then install a product, the files that result from that installation could be stored on the DECram disk. However, these files will be lost if the system fails or is taken out of service.

---

To prevent this loss, redefine the search list back to its former definition before installing the product. This deletes the DECram disk from the search list, as shown in the following example:

```
$ DEFINE/SYSTEM CONSTRUCTION "SYS$SYSDEVICE:[CONSTRUCTION]"
$ SHOW LOGICAL CONSTRUCTION -
_.$ "CONSTRUCTION" = "SYS$SYSDEVICE:[CONSTRUCTION]"
```

After the software installation completes, redefine the logical name to add the DECram disk back to the search list.

### 3.3 Shadow Sets of DECram Devices

Compaq recommends that shadow sets of DECram devices consist of at least two members. A volume shadow set consisting of a single member that is a DECram device is subject to the failure scenario detailed in Section 3.4.3.

Starting with DECram Version 3.0 software, DECram devices can be shadowed to real physical disks or partitions. This is useful for applications that rely on data integrity and also require enhanced performance. The Volume Shadowing for OpenVMS software will read from the DECram device, rather than from the physical device, thereby providing a performance advantage.



The Volume Shadowing for OpenVMS software will write to both the physical disk and DECram disk, so those writes will take longer than reads. The advantage of Volume Shadowing for OpenVMS software writing to both the physical disk and DECram disk is that data integrity is preserved. If a node fails, all data on the physical disk or a partition is still available and is not lost.

### 3.4 DECram Limitations

This section describes how using a DECram device can have a negative effect on system operation.

#### 3.4.1 Impact to Paging and Swapping

Using DECram disks reduces the size of the free page list. This may affect paging and swapping because of reduced available memory and could reduce or eliminate any performance gain in the I/O speed.

Some applications may require changing the values for the following system parameters for DECram disk memory requirements:

- FREELIM
- GROWLIM
- FREEGOAL

The following system parameters may need to be increased if you move installed images onto the DECram disk and reinstall them:

- GBLSECTIONS
- GBLPAGES

See the *OpenVMS System Management Utilities Reference Manual: M-Z* for information on system parameters.

#### 3.4.2 I/O Delay Characteristics Affecting DECram Use

Using a DECram disk alters the execution characteristics of the system and of the applications that use DECram disks for I/O operations. I/O operations that use DECram disks are CPU intensive; the I/O is essentially a movement of data from system memory to user memory that executes in the system context. Therefore, applications that implicitly rely on I/O delay characteristics are not recommended for DECram disk use.

In some cases, applications that do not consume large quantities of CPU resources because they are limited by I/O delay throughput will suddenly become very CPU-bound. The applications become CPU-bound because there is no waiting for I/O operations to complete. This change in CPU consumption could then affect other users on the system.

#### 3.4.3 Implications of Local Served and Shadowed DECram Devices

The failure characteristics of a DECram disk present an interesting problem in a served environment. The failure characteristics of a DECram disk are as follows:

- When the controller for the CPU where the memory allocated to the DECram disk resides fails, the media is destroyed as well.
- When the controller is reinitialized, the DECram disk devices that existed prior to system failure are no longer present.

## Using a DECram Disk

### 3.4 DECram Limitations

#### 3.4.3.1 Enhanced Operation

When any DECram device has memory allocated to it (that is, it is initialized with a size of non-zero), a record of this is made in the SYSSCOMMON:[SYSMGR]MDRECOVER.DAT file. Conversely, when the DECram device has memory released (that is, it is initialized to a size of zero), that memory deallocation is recorded in the same file.

During system initialization the SYSSCOMMON:[SYSMGR]MDRECOVER.DAT file is interrogated. If any records of DECram devices that had memory allocated to them are found, the I/O database for those devices is restored.

When the MOUNT verification process on the other OpenVMS Cluster members interrogates these restored DECram devices, the MOUNT verification terminates and the applications currently using the devices receive an error status for any outstanding I/Os. No further use of these devices, such as dismounting them clusterwide and reinitializing them on the serving OpenVMS Cluster member, can be made.

## A

---

- Allocating space, 2-1, 2-4
  - local disk
  - shared disk, 2-5
- Applications
  - identifying commonly used images, 3-2
  - locating files, 3-4
  - on workstations, 3-2
  - restrictions, 3-1
  - using DECram disk, 3-1

## C

---

- Comparison of a DECram disk
  - to disk caches, 1-2
- Configuration, 2-1
  - allocating memory, 2-4
  - creating
    - a DECram device, 2-3
  - deallocating memory, 2-5
  - determining device resources, 2-11
  - INITIALIZE command, 2-4
  - mounting, 2-14
  - mounting a shadowed DECram disk, 2-13
  - system resources, 2-2
- Configuring DECram disks
  - Version 2.3, 2-2
  - Version 3.0, 2-2
- CONNECT command, 2-3
- Controller failure
  - effects of, 3-6
- Creating
  - a DECram device, 2-3
- Creating and formatting RAM disks
  - INITIALIZE command, 2-1

## D

---

- Deallocating space, 2-5
  - local disk, 2-5
  - shared disk, 2-5
- DECram command
  - command interface, 2-1
- DECram disk
  - allocating space for, 2-4
  - characteristics, 1-1

- DECram disk (cont'd)
  - comparison to conventional disk, 1-2
  - comparison to disk caches, 1-2
  - configuration, 2-8
    - allocating memory, 2-4
    - creating a DECram device, 2-3
    - deallocating memory, 2-5
    - determining device resources, 2-11
    - system resources, 2-2
  - configuring, 2-1
  - deallocating space, 2-5
  - description, 1-1
  - disk size limitation, 2-5
  - following a dismount, 2-9
  - initialize
    - failure recovery, 2-5
  - MDDRIVER, 2-2
  - mounting, 2-8, 2-9
    - following a dismount, 2-9
  - MSCP served, 1-1
  - Multiple DECram disks, 2-1
  - recommended use of, 3-1
    - on workstations, 3-2
    - SYSSCRATCH files, 3-2
  - restrictions, 3-1
  - shadowing, 1-1
- DECram disk volume-label, 2-9
- DECram startup procedures, 2-1
- DECram volume shadow reads and writes, 3-4
- Disk space limit, 2-4

## F

---

- Failure characteristics, 3-5
- Failure recovery
  - INITIALIZE command, 2-5
- File lookups, 3-4
- Free page list
  - effect of reduced size, 3-5

## I

---

- I/O database, 2-2
  - creation of, 2-3
- I/O operations
  - delays in applications, 3-5

INITIALIZE command, 2-4  
    failure recovery, 2-5  
Install utility (INSTALL)  
    LIST command, 3-2  
IO CONNECT command  
    System Management (SYSMAN) utility, 2-3

## L

---

Limitations of DECram disk  
    I/O delays in applications, 3-5  
    paging and swapping behavior, 3-5  
    redefining system logical names, 3-4  
LIST command  
    Install Utility, 3-2  
Locating files, 3-4  
Logical names  
    redefining system, 3-4

## M

---

MDDRIVER, 2-2  
Memory  
    allocating, 2-4  
    deallocating, 2-5  
    determining device resources, 2-11  
    effect of reduced available, 3-5  
    system resources, 2-2  
Mounting a DECram disk  
    following a dismount, 2-9  
    local, 2-8, 2-9  
    shared, 2-8, 2-9  
Mounting a shadowed DECram disk, 2-14  
Mounting a shared DECram disk, 2-9  
MOUNT verification process, 3-6  
MSCP serving a DECram disk, 1-1

## N

---

/nocache qualifier, 2-13

## O

---

OpenVMS cluster configurations, 1-1  
    serving a DECram disk, 1-1  
    shadowing a DECram disk, 1-1

## P

---

Paging and swapping behavior, 3-5  
Privileges required, 2-5

## R

---

Read/write disk  
    restriction on use, 1-1

Recovering lost files, 3-3  
Recovery files  
    DECRAMSRECOVER.DAT  
    MDRECOVER.DAT, 2-11  
Redefining system logical names, 3-4  
Restoring a DECram disk, 2-10  
    after a power failure, 2-10  
    after a system crash, 2-10  
Restoring a disk, 2-10  
Restrictions, 3-1, 3-4  
RMS AI Journaling  
    using to recover DECram files, 3-3

## S

---

Search list, 3-4  
Search order, 3-4  
Served DECram disks  
    failure characteristics, 3-5  
Shadowed DECram disks, 2-13  
    clusters, 2-2  
    configuration, 2-13  
    dismounting, 2-13  
    mounting, 2-13  
Shadowing a DECram disk, 3-5  
Shadowing to a physical disk, 2-2  
Shadow Set  
    number of members, 3-4  
SHOW DEVICE command, 2-11  
SHOW DISK command  
    deleting .DAT files, 2-11  
    displaying disk information, 2-11  
/SIZE qualifier, 2-4  
Startup files, 2-10  
SYSSCRATCH files  
    storing on a DECram disk  
    improving system performance, 3-2  
SYSMAN command  
    command interface, 2-1  
System idle time, 3-5  
System Management (SYSMAN) utility  
    IO CONNECT command, 2-3  
System parameters  
    FREEGOAL, 3-5  
    FREELIM, 3-5  
    GBLPAGES, 3-5  
    GBLSECTIONS, 3-5  
    GROWLIM, 3-5  
    requiring modification, 3-5  
System resources  
    determining, 2-11  
    memory, 2-2  
System Page Table Entries (SPTes), 2-2

## **U**

---

Using a DECram disk, 3-1

## **W**

---

Workstations

    using DECram disk with  
    diskless workstations, 3-2

