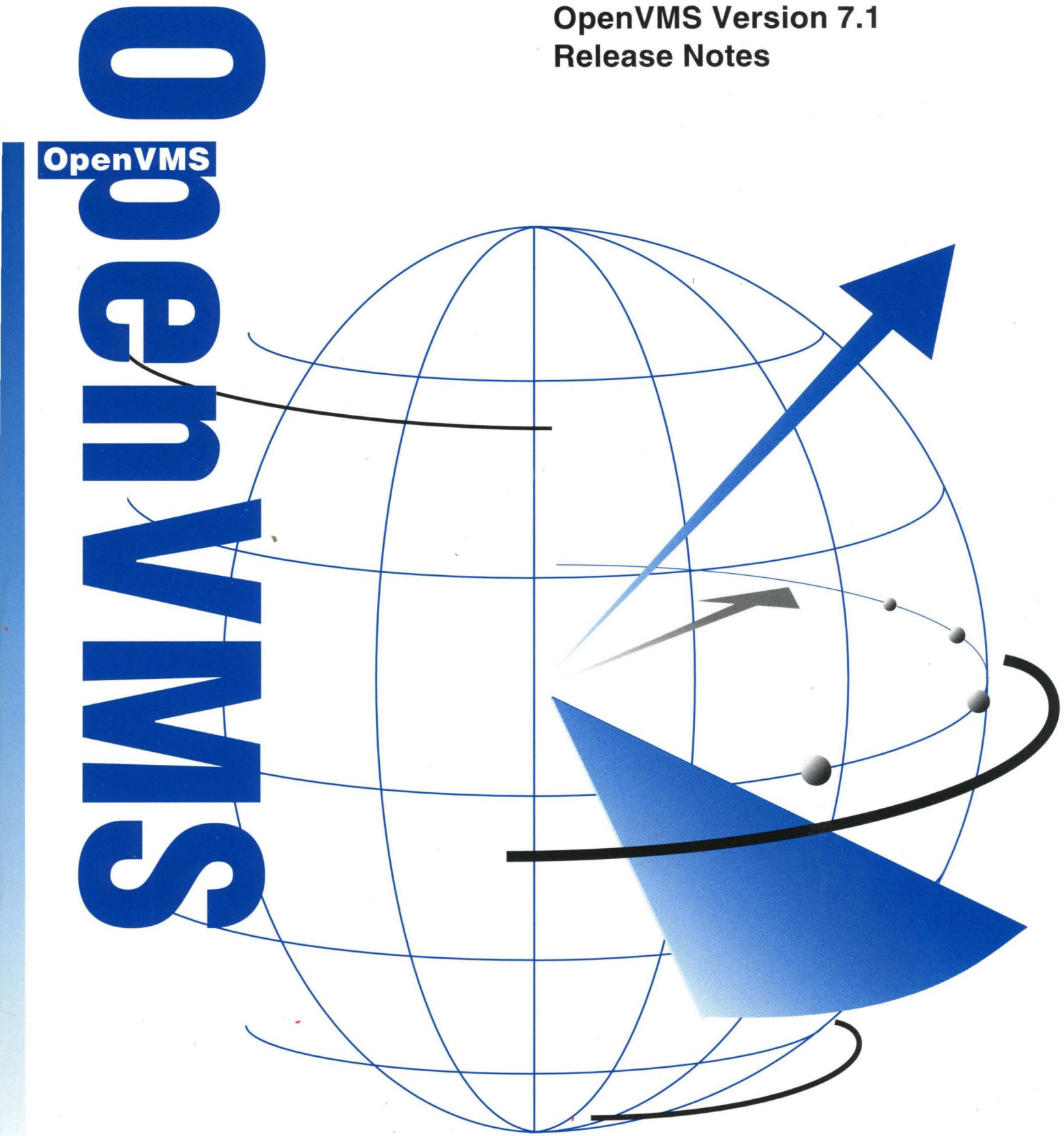
digital

OpenVMS Version 7.1
Release Notes

OpenVMS



OpenVMS Version 7.1 Release Notes

Order Number: AA–QSBTB–TE

November 1996

This manual describes changes to the software; installation, upgrade, and compatibility information; new and existing software problems and restrictions; and software and documentation corrections.

Revision/Update Information:	This is a new manual.
Software Version:	OpenVMS Alpha Version 7.1 OpenVMS VAX Version 7.1

Digital Equipment Corporation
Maynard, Massachusetts

November 1996

ZK6481

The OpenVMS documentation set is available on CD–ROM.

Contents

3 General User Release Notes

4 System Management Release Notes

5 Programming Release Notes

6 Device Support on OpenVMS Systems

A OpenVMS Alpha Version 6.2—1H3 Device Naming Notes

B Remedial Kits Included in OpenVMS Version 7.1

C Product Retirement Notices

Index

Figures

Tables

Preface

Intended Audience

This manual is intended for all OpenVMS operating system users. Read this manual before you install, upgrade, or use Version 7.1 of the operating system.

Document Structure

This manual contains the following chapters and appendixes:

- Chapter 1 contains release notes that pertain to installing and upgrading the OpenVMS operating systems, as well as hardware-related information.
- Chapter 2 contains installation and support information about OpenVMS layered products.
- Chapter 3 contains release notes about the general use of the OpenVMS operating system.
- Chapter 4 contains release notes specific to system management issues.
- Chapter 5 contains release notes that relate to programming on an OpenVMS system, including notes for compilers, linkers, and run-time library routines.
- Chapter 6 contains release notes pertaining to OpenVMS device support on Alpha and VAX systems.
- Appendix A contains device naming notes from OpenVMS Alpha Version 6.2—1H3.
- Appendix B lists remedial kits that are included in OpenVMS Version 7.1.
- Appendix C contains information about OpenVMS products that are no longer supported or that are slated for retirement. It also lists manuals that have been archived with this release.

In Chapters 2 through 5, notes are organized by facility or product name; facilities and products are listed alphabetically. Notes are further classified as changes, problems or restrictions, corrections, or documentation corrections.

This manual contains release notes introduced in the current release and notes from previous OpenVMS versions that still apply to the new release. Margin notes for each release note indicate the version of origin (for example, *V6.2*).

Notes from previous releases are published when:

- The release note has not been documented in hard copy in any other manual in the OpenVMS documentation set, and the note is still pertinent.

- The release note may be pertinent in multiple-version OpenVMS Cluster systems.

Note

This hardcopy version of the *OpenVMS Version 7.1 Release Notes* includes the following notes that were not published in the version of this manual that ships on the OpenVMS Version 7.1 CD–ROMs: Section 3.1.3.1, Section 4.15.2.5.2, Section 4.24.2.1, and Section 6.7. A few Alpha kits have also been added to the list in Appendix B.

Related Documents

For a list of additional documents that are available in support of this version of the OpenVMS operating system, refer to the *Overview of OpenVMS Documentation*.

For additional information on the Open Systems Software Group (OSSG) products and services, access the Digital OpenVMS World Wide Web site. Use the following URL:

`http://www.openvms.digital.com`

Reader's Comments

Digital welcomes your comments on this manual.

Print or edit the online form SYS$HELP:OPENVMSDOC_COMMENTS.TXT and send us your comments by:

Internet	**openvmsdoc@zko.mts.dec.com**
Fax	603 881-0120, Attention: OSSG Documentation, ZKO3-4/U08
Mail	OSSG Documentation Group, ZKO3-4/U08 110 Spit Brook Rd. Nashua, NH 03062-2698

How To Order Additional Documentation

Use the following table to order additional documentation or information. If you need help deciding which documentation best meets your needs, call 800-DIGITAL (800-344-4825).

Telephone and Direct Mail Orders

Location	Call	Fax	Write
U.S.A.	DECdirect 800–DIGITAL 800–344–4825	Fax: 800–234–2298	Digital Equipment Corporation P.O. Box CS2008 Nashua, NH 03061
Puerto Rico	809–781–0505	Fax: 809–749–8300	Digital Equipment Caribbean, Inc. 3 Digital Plaza, 1st Street, Suite 200 P.O. Box 11038 Metro Office Park San Juan, Puerto Rico 00910–2138
Canada	800–267–6215	Fax: 613–592–1946	Digital Equipment of Canada, Ltd. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6 Attn: DECdirect Sales
International	—	—	Local Digital subsidiary or approved distributor
Internal Orders	DTN: 264–4446 603–884–4446	Fax: 603–884–3960	U.S. Software Supply Business Digital Equipment Corporation 8 Cotton Road Nashua, NH 03063–1260

ZK–7654A–GE

Conventions

The name of the OpenVMS AXP operating system has been changed to the OpenVMS Alpha operating system. Any references to OpenVMS AXP or AXP are synonymous with OpenVMS Alpha or Alpha.

VMScluster systems are now referred to as OpenVMS Cluster systems. Unless otherwise specified, references to OpenVMS Clusters or clusters in this document is synonymous with VMSclusters.

In this manual, every use of DECwindows and DECwindows Motif refers to DECwindows Motif for OpenVMS software.

The following conventions are also used in this manual:

Ctrl/*x*	A sequence such as Ctrl/*x* indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.
PF1 *x* or GOLD *x*	A sequence such as PF1 *x* or GOLD *x* indicates that you must first press and release the key labeled PF1 or GOLD and then press and release another key or a pointing device button. GOLD key sequences can also have a slash (/), dash (–), or underscore (_) as a delimiter in EVE commands.
Return	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.)

. . .	Horizontal ellipsis points in examples indicate one of the following possibilities: • Additional optional arguments in a statement have been omitted. • The preceding item or items can be repeated one or more times. • Additional parameters, values, or other information can be entered.
. . .	Vertical ellipsis points indicate 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, if you choose more than one option, you must enclose the choices in parentheses.
[ ]	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, braces indicate a required choice of options; you must choose one of the options listed.
text style	This text style represents the introduction of a new term or the name of an argument, an attribute, or a reason. This style is also used to show user input in Bookreader versions of the manual.
italic text	Italic text indicates important information, complete titles of manuals, or variables. Variables include information that varies in system output (Internal error *number*), in command lines (/PRODUCER=*name*), and in command parameters in text (where *device-name* contains up to five alphanumeric characters).
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 type`	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 radixes—binary, octal, or hexadecimal—are explicitly indicated.

1

OpenVMS Installation, Upgrade, and Hardware Release Notes

This chapter contains information that applies to installations and upgrades of the OpenVMS VAX and OpenVMS Alpha operating systems. It also provides information specific to certain hardware.

The installation and upgrade notes in this chapter are organized into the following categories:

- Installation and upgrade notes common to both VAX and Alpha systems (see Section 1.2)
- VAX specific installation and upgrade notes (see Section 1.3)
- Alpha specific installation and upgrade notes (see Section 1.4)

Hardware and firmware notes follow the upgrade and installation sections.

For information about layered product installation and support, see Chapter 2.

1.1 Digital's Support Policy

Digital has recently announced a revised support policy and a new Prior Version Support Service for customers concerned that Digital's release cycles and the end of support may not be in concert with their ability to migrate systems to the new software releases.

Under Digital's new policy, support will be available for the current version and the previous version (for up to 12 months) when a customer purchases a standard support contract.

For versions of Digital software older than the previous version, Prior Version Support service is now available in two levels of support, including full, remedial support on selected products and versions.

For more information about all levels of support, contact your Digital support representative.

1.2 Installation and Upgrade Information Common to VAX and Alpha

The following release notes document installation and upgrade information common to both platforms. For more VAX specific installation and upgrade notes, see Section 1.3. For additional Alpha specific notes, see Section 1.4.

1.2.1 Changes and Enhancements

This section describes information related to installing or upgrading the OpenVMS VAX and OpenVMS Alpha operating systems.

1.2.1.1 Networking Options

V7.1

OpenVMS provides customers with the flexibility to choose the network protocol of their choice. Whether you require TCP/IP, OSI, or DECnet, OpenVMS allows you to choose the protocol or combination of protocols that works best for your network. OpenVMS supports Digital and third-party networking products.

- **OpenVMS Alpha Version 7.1**

  Starting with OpenVMS Alpha Version 7.1, you can choose either, both, or neither of the following network options during the main operating system installation:

  - Digital TCP/IP Services for OpenVMS Version 4.1
  - DECnet-Plus Version 7.1 (formerly named DECnet/OSI)

- **OpenVMS VAX Version 7.1**

  Starting with OpenVMS VAX Version 7.1, you can choose to install DECnet-Plus Version 7.1 (formerly named DECnet/OSI) during the main operating system installation. If you wish to install Digital TCP/IP Services for OpenVMS Version 4.1, see Section 1.2.1.3 and the *Digital TCP/IP Services for OpenVMS Installation and Configuration* manual for information on how to install TCP/IP as a layered product.

DECnet-Plus (Phase V) has replaced DECnet for OpenVMS (Phase IV) in the operating system installation and upgrade procedures for OpenVMS VAX and Alpha Version 7.1. DECnet-Plus contains all Phase IV functionality, plus the ability to run over TCP/IP or OSI protocols. When you choose DECnet-Plus, the product is automatically installed. After you complete your installation or upgrade of OpenVMS, you run the DECnet-Plus configuration program that prompts you for answers and quickly sets up your configuration. The configuration program has three options: Fast, Basic, or Advanced. See the *DECnet-Plus for OpenVMS Installation and Basic Configuration* manual for information on which option is best for you.

While most customers will find the upgrade to DECnet-Plus to be problem free, customers should consider their system capacity and performance before upgrading. First, customers should refer to the OpenVMS Software Product Description (SPD) to make sure they have sufficient memory and disk capacity to support the upgrade to Version 7.1. The SPD lists requirements for VAX and Alpha systems running various combinations of the operating system, DECwindows, and either DECnet-Plus or DECnet for OpenVMS. Second, customers running systems with high CPU utilization (an average of 75% or higher) or customers whose network server is near capacity may want to consider remaining with Phase IV. On some heavily loaded systems, the increased CPU requirements of DECnet-Plus and other layered products may cause a decline in system performance. Using available system management and network monitoring tools, you can measure your system and network performance prior to performing the upgrade.

Because establishing a distributed namespace, using either DECdns or the DNS/BIND options, is a complex procedure requiring coordination with corporate networking managers, Digital recommends that you do not move to distributed naming at the same time you upgrade your operating system and your networking product to DECnet-Plus. Use the default Local namespace and make sure that both your operating system and DECnet-Plus upgrade are successful and stable before introducing a distributed namespace. Refer to the *DECnet-Plus Planning Guide* for a discussion on how to plan and manage the move to a distributed namespace.

If you prefer to install DECnet for OpenVMS (Phase IV), you can install it as a layered product. On Alpha systems, you can install Phase IV software from the layered product menu. Refer to the *OpenVMS Alpha Version 7.1 Upgrade and Installation Manual*. On VAX systems, you can install the Phase IV software from a kit provided on the distribution medium. Refer to the *OpenVMS VAX Version 7.1 Upgrade and Installation Manual*. See also the *DECnet for OpenVMS Guide to Networking* for complete instructions on how to install DECnet for OpenVMS (Phase IV) as a layered product. Note that DECnet for OpenVMS is now supported under the new Prior Version Support service (see Section 1.1).

1.2.1.2 DECnet Documentation

V7.1

To assist customers moving from DECnet Phase IV to DECnet-Plus, OpenVMS is delivering a one-time complimentary offering that includes DECnet-Plus software and a DECnet-Plus Starter Kit. Depending on the OpenVMS offering format, the DECnet-Plus Starter Kit includes documentation in either printed or online format. Refer to the *DECnet-Plus for OpenVMS Introduction and User's Guide* for further information on DECnet-Plus for OpenVMS manuals. In future releases, DECnet-Plus hardcopy manuals will not be part of the OpenVMS Documentation Set and must be purchased separately. However, they will continue to be included on the OpenVMS Documentation CD–ROM.

The DECnet for OpenVMS (Phase IV) software will ship on the OpenVMS Version 7.1 operating system media. The DECnet for OpenVMS manuals are no longer part of the hardcopy OpenVMS Documentation Set, but are included on the OpenVMS Documentation CD–ROM in both Bookreader and PostScript format. You can also order printed books separately through DECdirect (800-344-4825). The DECnet Phase IV manuals and their order numbers are as follows:

DECnet for OpenVMS Guide to Networking	AA–PV5ZA–TK
DECnet for OpenVMS Networking Manual	AA–PV60A–TK
DECnet for OpenVMS Network Management Utilities	AA–PV61A–TK

1.2.1.3 Digital TCP/IP Services for OpenVMS Installation Requirement

V7.1

Digital TCP/IP Services for OpenVMS Version 4.1 software is included on the OpenVMS Alpha and OpenVMS VAX Version 7.1 CD–ROMs. The Digital TCP/IP Kit along with a mandatory Security Update Kit are available in the following directories on the Alpha and VAX CD–ROMs:

- [TCPIP_ALPHA_041] (for Alpha)
- [TCPIP_VAX_041] (for VAX)

If you choose to install the Digital TCP/IP Services for OpenVMS software, you must install both kits:

1. Install the appropriate Version 4.1 Kit during the OpenVMS operating system installation or afterward, as a layered product.
2. Install the appropriate mandatory Security Update Kit, which adds important security functionality to the Version 4.1 software.

1.2.1.4 OpenVMS Cluster Compatibility Kit for Version 6.2 Systems

V7.1

The OpenVMS Cluster Compatibility Kit provides many OpenVMS Version 7.1 enhancements for Version 6.2 systems. This kit is required for Version 6.2 systems if they are included in a cluster with Version 7.1 systems (same system architecture or a mix of VAX and Alpha systems). Optionally, users can install it on other OpenVMS Version 6.2 systems to derive the same benefits.

Cluster Compatibility Kit Features

The Cluster Compatibility Kit includes the following improvements for systems running OpenVMS Version 6.2:

- OpenVMS Version 7.1 Volume Shadowing enhancements

  The volume shadowing enhancements include significant quality improvements and an increase in supported shadow set members from 400 to 500. Note that the Version 7.1 volume shadowing system disk minimerge feature is not included in the Cluster Compatibility Kit nor is the Dump file off the system disk for OpenVMS Alpha. (Dump file off the system disk has been available for OpenVMS VAX systems since Version 6.2.)

  Note

  If you use volume shadowing, be sure to read the volume shadowing release notes in Section 4.24.

- OpenVMS Version 7.1 Mount enhancements

  The Mount utility has been completely rewritten, resulting in a faster, more robust utility.

- Correction to a lock manager problem found in OpenVMS Version 6.2

  The lock manager changes correct a problem in OpenVMS Version 6.2 that could corrupt some internal states in lock information used by fork lock routines, notably the I/O cache subsystem. This problem was corrected in OpenVMS Version 7.0, and also in the limited hardware releases for OpenVMS Alpha Version 6.2-1H2 and Version 6.2-1H3.

- Limited support for new port allocation classes for SCSI devices

  Port allocation classes are a new naming option for SCSI devices on systems running OpenVMS Alpha Version 7.1. If you have installed the Cluster Compatibility Kit (on a VAX or Alpha system), you can access SCSI disks on an OpenVMS Alpha Version 7.1 system that use port allocation classes in their names, but you cannot name SCSI disks on a Version 6.2 system with port allocation classes.

Snapshot Facility Disabled

Installing the Cluster Compatibility Kit on a Version 6.2 system disables the Snapshot facility, which has been removed from OpenVMS VAX Version 7.1 (see Section C.9).

System Dump Analyzer Utility (SDA)

A special version of the OpenVMS Version 6.2 System Dump Analyzer (SDA) utility is included in the Cluster Compatibility Kit. It recognizes the new Volume Shadowing data structures.

When you install the Cluster Compatibility Kit, the existing OpenVMS Version 6.2 SDA is renamed SDA_OLD.EXE and the Cluster Compatibility Kit version is named SDA.EXE. Use SDA_OLD.EXE to analyze crash dumps from an OpenVMS Version 6.2 system that has not installed the Cluster Compatibility Kit. Use SDA.EXE to analyze crash dumps from an OpenVMS Version 6.2 system that has installed the Cluster Compatibility Kit.

Installing the Cluster Compatibility Kit

Two kits are available, one for OpenVMS VAX Version 6.2 systems and one for OpenVMS Alpha Version 6.2 systems. Each kit is on its respective OpenVMS CD–ROM, as follows:

- [CLUSTER_COMP_VAX062]VAXCOMPAT_062.A (for VAX)
- [CLUSTER_COMP_ALPHA062]ALPCOMPAT_062.A (for Alpha)

To install the VAX kit, use the following command:

```
$ @SYS$UPDATE:VMSINSTALL VAXCOMPAT_062.A ddcu:[CLUSTER_COMP_VAX062]
```

To install the Alpha kit, use the following command:

```
$ @SYS$UPDATE:VMSINSTALL ALPCOMPAT_062.A ddcu:[CLUSTER_COMP_ALPHA062]
```

1.3 Installation and Upgrade Information Specific to VAX

The release notes in this section pertain only to installations or upgrades of OpenVMS VAX operating systems. See Section 1.2 for additional notes that pertain to both VAX and Alpha systems. For complete information about installing or upgrading your OpenVMS VAX Version 7.1 operating system, refer to the *OpenVMS VAX Version 7.1 Upgrade and Installation Manual*.

1.3.1 Changes and Enhancements

The following note describes a change in magnetic tape distribution of the OpenVMS VAX operating system.

1.3.1.1 Magnetic Tape Distribution

V7.1

Distribution of the OpenVMS VAX operating system on magnetic tape is now available only to customers who subscribe to the OpenVMS VAX Media and Hardcopy Documentation Update Service.

OpenVMS VAX CD–ROM and TK50 kits are available from all of the following sources:

- DECdirect (800 344–4825)
- OpenVMS VAX Media and Hardcopy Documentation Update Service
- A Digital sales representative

- An authorized reseller

1.4 Installation and Upgrade Information Specific to Alpha

The release notes in this section pertain only to installations or upgrades of OpenVMS Alpha operating systems. See Section 1.2 for additional notes that pertain to both Alpha and VAX systems. For complete information about installing or upgrading your OpenVMS Alpha Version 7.1 operating system, refer to the *OpenVMS Alpha Version 7.1 Upgrade and Installation Manual*. An appendix in that manual also includes additional release notes about specific Alpha computers.

1.4.1 Changes and Enhancements

This section describes changes and enhancements to OpenVMS Alpha installation and upgrade procedures.

1.4.1.1 Enabling the DECevent DIAGNOSE Command

V7.1

In OpenVMS Alpha Version 7.0 and earlier releases of OpenVMS Alpha, the DECevent DCL command DIAGNOSE was defined during the operating system installation or upgrade.

Beginning with OpenVMS Alpha Version 7.1, the definition of the DIAGNOSE command during installation or upgrade is disabled. In order to enable the DIAGNOSE command in OpenVMS Version 7.1, the DECevent kit provided on the OpenVMS Alpha Version 7.1 CD–ROM must be installed following the installation of OpenVMS Alpha Version 7.1. For information about the location of the DECevent kit, see the *Guide to OpenVMS Version 7.1 CD–ROMs*.

If the DECevent kit provided on the OpenVMS Alpha CD–ROM is not installed after the operating system, users attempting to use the DIAGNOSE command will receive the following system message:

```
$ DIAGNOSE [parameters]
%DIA-E-NOINSTAL, DIAGNOSE has not been installed on this system
$
```

1.4.2 Problems and Restrictions

This section describes problems and restrictions that apply to installing or upgrading to OpenVMS Alpha Version 7.1.

1.4.2.1 Spiralog File System Support

V7.1

OpenVMS Alpha Version 7.1 requires a new version of the Spiralog file system. Customers running prior versions of Spiralog must upgrade to Spiralog Version 1.2, which is available on CD–ROM. Contact your Digital support representative to order the software and documentation, as follows:

Spiralog Version 1.2 Offerings	Order Number
CD–ROM with software and online documentation	QA–4P7AA–H8
Hardcopy documentation	QA–4P7AA–GZ

Warning

Before upgrading from OpenVMS Alpha Version 7.0, deinstall Spiralog. Once OpenVMS Alpha Version 7.1 has been installed, you can install Spiralog Version 1.2.

If you upgrade your OpenVMS Alpha Version 7.0 system to OpenVMS Alpha Version 7.1 when Spiralog is installed, and later remove Spiralog or install any version of Spiralog, this action will cause a failure in the Backup utility, the batch and print queuing system, and DECdtm services.

If you accidentally upgrade your OpenVMS Alpha system to Version 7.1 while Spiralog is installed, you must deinstall Spiralog, reinstall OpenVMS Alpha Version 7.1 using the PRESERVE option, install Spiralog Version 1.2, and reboot your system in order to recover.

Considerations When Choosing a File System

With OpenVMS Alpha you can choose to use the standard Files–11 file system or the new Spiralog file system for each disk in your configuration. By default, the Files–11 file system is used for all disks.

Before deciding to use Spiralog for any of your disks, carefully consider the following tradeoffs:

- Spiralog benefits:
  - High-speed backup and restore operations
  - Online backup
  - Scalability to support large numbers of files and very large files
- Temporary Spiralog restrictions:
  - Digital plans to modify the Spiralog on-disk structure significantly in a future release (within approximately 18 to 24 months). This change will require the complete backup, reinitialization, and restoration of all existing Spiralog volumes.
  - Tapes archived using the current version of Spiralog will not automatically restore on future versions of Spiralog.
  - It is vital to ensure that Spiralog volumes do not run out of space during normal operation. A full Spiralog disk becomes a read-only volume, where you cannot delete a file. Recovery requires a full backup and restore of the volume. Spiralog Version 1.2 reserves some free space and sets a capacity threshold on each volume. When that capacity threshold is reached, no further writes are permitted and the cleaner is invoked to regain free space and delay volumes from exceeding their capacity.
  - The Spiralog on-disk structure requires that disk volumes do not experience unrecoverable disk errors (that is, unrecoverable bad blocks). Bad blocks that occur in critical areas of the directory structure can render the entire volume unreadable. To overcome this limitation, it is necessary to deploy all Spiralog volumes on RAID–1 (volume shadowed) or RAID–5 protected disks.

  These issues will be resolved in a future Spiralog release.

If you decide to use Spiralog, your system environment must meet the prerequisites outlined in the Spiralog Version 1.*n* Software Product Description (SPD).

1.5 AlphaServer 4100 (Alpha Only)

This section contains release notes pertaining to the AlphaServer 4100.

1.5.1 Problems and Restrictions

The following note describes how to resolve an error that occurs on the AlphaServer 4100.

1.5.1.1 Field Replaceable Units (FRU) Table Error

V7.1

After you boot the OpenVMS Alpha operating system on an AlphaServer 4100, the following message might display on the screen:

```
*****Config packet buffer allocation failure: Continuing without writing Errorlog
```

This message indicates that the size of the FRU table exceeds the size of the buffer allocated by the default value of the SYSGEN parameter ERLBUFFERPAGES. It is a warning message only and indicates that the FRU table was not written in the error log on this reboot.

If your system displays this message, Digital recommends that you change the value of the ERLBUFFERPAGES parameter from 4 (the default) to 6. The following example shows how to use the SYSGEN utility to accomplish this task:

```
$ MCR SYSGEN
SYSGEN> USE CURRENT
SYSGEN> SET ERLBUFFERPAGES 6
SYSGEN> WRITE CURRENT
SYSGEN> EXIT
```

If this warning appears again after you reboot the system, increase the value of the ERLBUFFERPAGES parameter in increments of 2 (not exceeding the maximum of 32) until the warning message no longer displays. The value of ERLBUFFERPAGES that resolves the problem varies depending on the configuration of the system.

1.6 DEC 7000 (Alpha Only)

This section contains release notes pertaining to DEC 7000 systems.

1.6.1 Changes and Enhancements

The following note describes a change in the behavior of DEC 7000 systems.

1.6.1.1 Ctrl/P Behavior Change During Boot

V7.1

Starting with OpenVMS Alpha Version 7.1, the remote halt command, issued by typing Ctrl/P at the system console, does not work during boot until the following copyright banner appears.

```
$!  Copyright (c) 1996 Digital Equipment Corporation.  All rights reserved.
```

In previous versions of OpenVMS, typing Ctrl/P at the system console always returned the system to the console prompt at any time during the boot.

1.7 Qvision Graphics Board (Alpha Only)

See Section 2.10.1.1 for a behavior change that affects users with the Qvision graphics board on their workstation or server.

1.8 RF73 and Other RF*nn* DSSI Disk Devices

Release notes in this section pertain to the RF31T, RF31T+, RF35, RF35+, RF73, and RF74 DSSI disk devices.

1.8.1 Problems and Restrictions

This section describes a problem found in certain RF31T, RF31T+, RF35, RF35+, RF73, and RF74 DSSI disk devices.

1.8.1.1 RF73 and Other RF*nn* DSSI Disk Devices and Controller Memory Errors

V6.2

A problem exists with the microcode for earlier versions of RF31T, RF31T+, RF35, RF35+, RF73, and RF74 DSSI disk devices that can cause data loss. The problem can occur when reading data from one of these devices if the device has had a controller memory error (also known as an error detection and correction (EDC) error). The error could have been induced by a virtual circuit closure or faulty hardware.

Digital advises customers with any of these devices to check their microcode revision levels. If the microcode revision levels are lower than the numbers shown in Table 1–1, Digital recommends that you update the microcode. The microcode for all models, except the RF31T, RF31T+, and RF35+, is provided on the latest OpenVMS binary distribution CD–ROM.

The RF_VERS utility, a utility program that displays the microcode revision level of the DSSI disk devices, is also provided on the CD–ROM. Instructions for using the utility program and for updating the microcode are provided in this section.

Note

If you have an RF31T, RF31T+, or RF35+ disk drive with a version of microcode that is not supported (see Table 1–1), and if you have a support contract, contact your Digital support representative. Otherwise, contact your authorized reseller.

The earliest supportable revision levels of the DSSI disk microcode are shown in Table 1–1.

Table 1–1 Supported Microcode Revision Levels

Device Type	Minimum Level with Supported Microcode
RF31T	T387E
RF31T+	T387E
RF35	T392D
RF35+	T392D
RF36	V427P
RF73	T392D
RF74	V427P

To display the microcode version level of your DSSI disk devices, perform the following steps:

1. Log in to the SYSTEM account or another account that has the CMKRNL, DIAGNOSE, and SYSPRV privileges.
2. Issue the following commands:

```
$ SET PROCESS /PRIVILEGE=(DIAGNOSE,CMKRNL,SYSPRV)
$ SHOW DEVICE FYA0:
```

   On VAX systems, if the SHOW DEVICE command produces an error, issue the following commands:

```
$ RUN SYS$SYSTEM:SYSGEN
        SYSGEN>  CONN FYA0/NOADAP
        SYSGEN>  ^Z
```

   On Alpha systems, if the SHOW DEVICE command produces an error, issue the following commands:

```
$ RUN SYS$SYSTEM:SYSMAN
        SYSMAN> IO CONNECT FYA0: /NOADAP
        SYSGEN>  ^Z
```

3. On VAX and Alpha systems, issue the following command:

```
$ RUN SYS$ETC:RF_VERS.EXE
```

The following is an example of the display produced by the RF_VERS utility:

```
Program Name:   RF_VERS
Revision Level: V1.2s

NOTICE: This program does not currently support the RF72 or any
        HSDxx controllers. See next version for support.

DSSI disks currently on this system as seen by RF_VERS

Device          Node        Status       Hardware    Firmware
Name            Name                     Type        Version

_$22$DIA7:      R4JL2I      mounted      RF73        T387A
_$22$DIA6:      R4I0BG      mounted      RF73        T387A
_$22$DIA8:      R4XLWE      mounted      RF73        T387A
_$22$DIA2:      R4FCZK      mounted      RF73        T387A
_$22$DIA3:      R4CKCG      mounted      RF73        T387A
_$22$DIA4:      R4ZKUE      mounted      RF73        T387A
_$22$DIA9:      R4GYYI      mounted      RF73        T387A
_$22$DIA1:      R4XRYI      mounted      RF73        T387A
```

To update the microcode in your device, use the appropriate command for your device and platform from Table 1–2.

Caution

Back up the disk before updating the microcode.

Table 1–2 Commands for Updating Microcode in Certain DSSI Disk Devices

Device Type	Platform	Command
RF35	Alpha	$RUN SYS$ETC:RF35_T392F_DEC_ALPHA.EXE
RF35	VAX	$RUN SYS$ETC:RF35_T392F_DEC.EXE
RF36	Alpha	$RUN SYS$ETC:RF36_V427P_DEC_ALPHA.EXE
RF36	VAX	$RUN SYS$ETC:RF36_V427P_DEC.EXE
RF73	Alpha	$RUN SYS$ETC:RF73_T392F_DEC_ALPHA.EXE
RF73	VAX	$RUN SYS$ETC:RF73_T392F_DEC.EXE
RF74	Alpha	$RUN SYS$ETC:RF74_V427P_DEC_ALPHA.EXE
RF74	VAX	$RUN SYS$ETC:RF74_V427P_DEC.EXE

Caution

Do not delete SCSI_INFO.EXE, RF_VERS.EXE, or any of the files listed in Table 1–2. If these files are deleted, VMSKITBLD.COM (on VAX) will not be able to find them. Similarly, on Alpha systems, the PRODUCT INSTALL commands in AXPVMS$PCSI_INSTALL and AXPVMS$PCSI_INSTALL_MIN will fail.

2

OpenVMS Layered Products Release Notes

This chapter contains installation and support information about OpenVMS layered products. Notes about using compilers, linkers, and run-time library routines are included in Chapter 5.

2.1 Layered Product Support

Information about layered product support is available in the Software Public Rollout Reports for OpenVMS, available on the World Wide Web. The Software Public Rollout Reports for OpenVMS list the availability of Digital's software products shipping on the Software Products Library kits (CD–ROM consolidations) for OpenVMS Alpha and OpenVMS VAX.

For OpenVMS Version 7.1, three new reports show currently shipping and previously released Digital layered software products for the following operating system versions:

- OpenVMS Version 6.2 only
- OpenVMS Version 6.2 and higher
- OpenVMS Version 7.0 and higher

The reports contain the product name and version, the operating system version required to support the product, and the volume ship date for the product. The information in these tables is continually evolving and is subject to change. The reports are intended for public distribution and are updated monthly. The information is not provided in these release notes because of the changing nature of the information.

These reports are available from the OpenVMS home page on the World Wide Web in the OpenVMS Products section. Use the following URL to access the OpenVMS Software Public Rollout Reports for OpenVMS:

```
http://www.openvms.digital.com/openvms/os/swroll.html
```

If you do not have Internet access, you can find the operating system support information on any of the quarterly Software Products Libraries, in the following directory:

```
[README]SW_COMPAT_MATRIX.PS (.TXT)
```

The Software Public Rollout Reports are also available from your Digital support representative.

2.2 DEC BASIC

This section contains release notes pertaining to DEC BASIC.

2.2.1 Problems and Restrictions

The following note describes a build restriction for DEC BASIC.

2.2.1.1 BASIC$STARLET.TLB Build Restriction (Alpha Only)

V7.0

The introduction of 64-bit support for system services prevents the proper construction and use of BASIC$STARLET.TLB on OpenVMS Alpha systems.

If you do not need any of the system definitions from STARLET, take the default for the following question when installing DEC BASIC:

```
Do you want to install the OpenVMS AXP system definitions (10 min.) [NO]?
```

If you do need system definitions from STARLET, there are two possible workarounds:

- If you need only a few definitions, you can construct your own include library using the STARLET definitions as a guide.
- If you require a number of system definitions, you must construct your own copy of BASIC$STARLET.TLB, as follows:

  1. Before installing DEC BASIC, make a copy of STARLETSD.TLB in SYS$LIBRARY. Be sure no one uses STARLETSD.TLB until you have completed building BASIC$STARLET.TLB.
  2. Edit your copy of SYS$LIBRARY:STARLETSD.TLB to remove any system definitions that refer to 64-bit integers by *value*. (DEC BASIC does not support these, so you will not need them.)
  3. Install DEC BASIC and request that the system definitions be installed.
  4. When the DEC BASIC installation is complete, delete your copy of SYS$LIBRARY:STARLETSD.TLB. A usable copy of BASIC$STARLET.TLB will be in SYS$LIBRARY.

     It is now safe for other users to access STARLETSD.TLB.

2.3 DEC C and DEC C++

This section contains release notes pertaining to DEC C and DEC C++.

2.3.1 Changes and Enhancements

The following note describes a change in how STARLET header files now ship on OpenVMS VAX.

2.3.1.1 STARLET Header Files Now Ship With OpenVMS VAX

V7.1

Starting with Version 7.1, OpenVMS VAX directly supplies the STARLET header files for DEC C and DEC C++ in SYS$LIBRARY:SYS$STARLET_C.TLB, as has always been done on OpenVMS Alpha systems. DEC C and DEC C++ compiler Versions 5.2 or higher are required to access the STARLET headers. See a warning about installing older DEC C and DEC C++ compiler versions on OpenVMS VAX Version 7.1 in Section 2.3.2.1.

The content of the STARLET headers has also been edited to correct deficiencies in versions supplied by the DEC C and DEC C++ compilers in releases prior to Version 7.1.

2.3.2 Problems and Restrictions

The following notes describe problems associated with DEC C and DEC C++.

2.3.2.1 Pre-Version 5.2 Kits May Delete SYS$STARLET_C.TLB (VAX Only)

V7.1

Installing a version of the DEC C or DEC C++ compiler older than Version 5.2 on OpenVMS VAX Version 7.1 may damage or delete SYS$LIBRARY:SYS$STARLET_C.TLB. (See Section 2.3.2.2 for another warning about installing DEC C++ Version 5.3 on OpenVMS VAX Version 7.1.)

2.3.2.2 DEC C++ Version 5.3 Installation Fails (VAX Only)

V7.1

When you attempt to install DEC C++ Version 5.3 on VAX systems running OpenVMS Version 7.1, the installation fails because the Version 5.3 kit fails to install the system headers on OpenVMS Version 7.1.

DEC C++ Version 5.4 fixes these problems.

2.4 DEC Pascal

This section contains release notes pertaining to DEC Pascal.

2.4.1 Problems and Restrictions

The following notes concern reinstallation of DEC Pascal after OpenVMS Alpha is upgraded to Version 7.1.

2.4.1.1 Installing DEC Pascal After An Upgrade (Alpha Only)

V7.1

After upgrading to OpenVMS Alpha Version 7.1, you should reinstall DEC Pascal to produce new versions of STARLET.PAS and other definition files to match the upgraded system.

If you do not reinstall DEC Pascal after upgrading to OpenVMS Alpha Version 7.1, the compiler on your system will still work correctly. However, STARLET.PAS and the other definition files will not contain any of the new definitions added in Version 7.1.

Note that because of changes in OpenVMS, the DEC Pascal Version 5.5 kit can sometimes go into an infinite loop when it is installed on OpenVMS Alpha Version 7.1. A new Pascal patch kit solves this problem. Contact your Digital support representative to get the kit. The kit will also be included in the February 1997 Software Products Library for OpenVMS Alpha.

2.5 DEC PL/I (Alpha Only)

This section contains release notes pertaining to DEC PL/I for OpenVMS Alpha.

2.5.1 Problems and Restrictions

The following note describes a DEC PL/I restriction.

2.5.1.1 RTL Support for OpenVMS Version 7.1

V7.1

If you have installed DEC PL/I Version 4.1 for OpenVMS Alpha and use the DEC PL/I run-time libraries, you must use the latest RTL that ships with the DEC PL/I Version 4.1 kit. (Note that the latest RTL does not ship with Version 7.1 of the operating system.) The DEC PL/I Version 4.1 RTL does not support OpenVMS Alpha Version 7.1; it supports only OpenVMS Alpha Versions 6.2 and 7.0. However, PL/I code compiled on prior versions of DEC PL/I will continue to run on OpenVMS Alpha Version 7.1.

2.6 DECforms

This section contains release notes pertaining to DECforms.

2.6.1 Problems and Restrictions

The following note describes a DECforms support issue.

2.6.1.1 Support on OpenVMS Version 7.0 and Later (Alpha Only)

V7.0

Because of changes in DECthreads, DECforms Version 2.1 does not work with OpenVMS Alpha Version 7.0 and later. Installing OpenVMS Alpha Version 7.0 or later causes existing applications based on DECforms Version 2.1 to fail; installing DECforms Version 2.1 on OpenVMS Alpha Version 7.0 or later also fails. In both cases, you get the following error message:

```
%CMA-F-USE_ERROR, requested operation is inappropriate for the
specified object
```

In order for DECforms based applications to operate correctly on OpenVMS Alpha Version 7.0 and later, you must run DECforms Version 2.1A or later.

2.7 DECpresent

This section contains a release note pertaining to installing DECpresent.

2.7.1 Problems and Restrictions

The following note describes a dependence for installing DECpresent.

2.7.1.1 Installation Dependency on OpenVMS VAX Version 6.1 or Later

V6.1

To run DECpresent Version 1.0A on OpenVMS VAX Version 6.1 or later, you must upgrade the CDA Converter Library from Version 1.1 to Version 2.0.

When installing DECpresent Version 1.0A on OpenVMS VAX Version 6.1 or later, system managers can safely ignore the IVP failure for the CDA Converter Library Version 1.1 because that version of the product is bundled with DECpresent but does not work on OpenVMS VAX Version 6.1 and later.

After installing DECpresent Version 1.0A on OpenVMS VAX Version 6.1 or later, or upgrading from VMS Version 5.5–2 to Version 6.1 or later with DECpresent Version 1.0A already installed on the system, system managers should install CDA Converter Library Version 2.0.

2.8 DECram

This section contains a release note about DECram support.

2.8.1 Problems and Restrictions

The following note describes a DECram support limitation.

2.8.1.1 DECram Version 2.2 Is Not Supported (Alpha Only)

V7.0

You cannot install DECram Version 2.2 on OpenVMS Alpha Version 7.0 or later. Version 7.0 and later of OpenVMS Alpha supports only DECram Version 2.2B.

You also cannot upgrade from OpenVMS Alpha Version 6.2 to Version 7.0 or later if you have DECram Version 2.2 installed. Attempting to do so prevents the system from booting. A workaround that allows the upgrade to proceed is to use the DCL command RENAME to rename the DECram executable image, for example:

```
$ RENAME SYS$SPECIFIC:[SYS$LDR]DECRAM$EXECLET.EXE -
_$ SYS$SPECIFIC:[SYS$LDR]OLD_DECRAM$EXECLET.EXE
```

After renaming the executable image, you can install DECram Version 2.2B.

2.9 DECwindows Motif for OpenVMS

This section contains release notes pertaining to the DECwindows Motif for OpenVMS layered product.

2.9.1 Changes and Enhancements

This section contains a note about support for the DECwindows Motif for OpenVMS layered product.

2.9.1.1 DECwindows Motif Version 1.2 for OpenVMS No Longer Supported

V7.1

Starting with OpenVMS Version 7.1, the OpenVMS operating system no longer supports DECwindows Motif Version 1.2 (or earlier) for OpenVMS. To use DECwindows with OpenVMS Version 7.1, you must install DECwindows Motif Version 1.2–3 or later. If you use the English language version of DECwindows Motif, Digital recommends that you install Version 1.2–4. If you want to install a language variant, see Section 2.9.2.1.

DECwindows Motif Version 1.2–3 and later does provide run-time support for programs built on earlier versions of DECwindows and DECwindows Motif. For more information, see the *DECwindows Motif for OpenVMS Release Notes* for Version 1.2–3 or later.

For OpenVMS Alpha systems, DECwindows Motif Version 1.2–3 is supported *only* if you install a remedial kit. Remedial kits are also available for OpenVMS VAX systems. For information about remedial kits for both platforms, see Section 2.9.2.2.

2.9.2 Problems and Restrictions

This section describes problems and restrictions associated with the DECwindows Motif for OpenVMS layered product.

2.9.2.1 Language Variants for DECwindows Motif Version 1.2–4

V7.1

Not all language variants are available for DECwindows Motif Version 1.2–4. Consult your Digital support representative for more information.

2.9.2.2 Remedial Kits for DECwindows Motif Version 1.2–3

V7.1

For OpenVMS Alpha Version 7.1 systems, DECwindows Motif Version 1.2–3 is supported *only* if you install the appropriate remedial kit. OpenVMS VAX Version 7.1 supports DECwindows Motif Version 1.2–3, but it also has problems unless a remedial kit is installed.

Why You Should Install the Remedial Kit

If you run DECwindows Motif Version 1.2–3 on an OpenVMS Version 7.1 system and do not install one of the remedial kits described later in this note, OpenVMS Alpha users will experience all of the following problems and VAX users will experience all but the first three of these problems:

- The DECwindows login program fails with an ACCVIO status, preventing users from logging in to DECwindows.
- The DECwindows interface to DEC Notes fails to open remote notes conferences and displays the following error message:

```
LinkWorks Reported Error: Unknown error
```

- The following error message is displayed on nonworkstation systems during system startup:

```
%SDA-E-NOTINPHYS, 00000024 : virtual data not in physical memory
```

- Console broadcasts are disabled by default on nonworkstation systems (see Section 2.9.2.3).
- DECwindows system files are purged during startup (see Section 2.9.2.4).
- The locale support in Xlib is not compatible with the support in the DEC C Run-Time Library (see Section 5.5.2.1).

The remedial kit you install depends upon whether you are running the U.S. version of DECwindows Motif Version 1.2–3 or the worldwide version.

Remedial Kit for U.S. Version

If you want to use any of the following language variants, you must install the U.S. version of DECwindows Motif:

- French
- German
- Hebrew
- Italian
- Spanish
- Swedish

If you use the U.S. version of DECwindows Motif Version 1.2–3, install the following remedial kits, which ship on the OpenVMS Version 7.1 operating system CD–ROM and are also available from your Digital support representative:

- For VAX systems, VAXMOTF07_U3012

- For Alpha systems, ALPMOTF07_U3012

Remedial Kit for Worldwide Version

The following language variants include the worldwide version of DECwindows Motif:

- Czech
- Hangul
- Hanyu
- Hanzi
- Hungarian
- Japanese
- Polish
- Russian
- Slovak
- Thai

If you use the worldwide version of DECwindows Motif Version 1.2–3, install the following remedial kits, which ship on the OpenVMS Version 7.1 operating system CD–ROM and are also available from your Digital support representative:

- For VAX systems, VAXDWMW01_U3012
- For Alpha systems, ALPDWMW01_U3012

2.9.2.3 Console Broadcasts Disabled

V7.1

In the U.S. version of DECwindows Motif Version 1.2–3, console broadcasts are disabled by default by the DECwindows startup procedure on nonworkstation systems as well as on workstation systems. This problem is corrected in Version 1.2–4.

To allow broadcasts to OPA0: in the U.S. version of DECwindows Motif Version 1.2–3, edit the file SYS$MANAGER:DECW$PRIVATE_APPS_SETUP.COM (creating it if it does not exist) and add the following global symbol definition:

```
$ DECW$CONSOLE_SELECTION == "ENABLE"
```

Then, restart DECwindows by entering the following command:

```
$ @SYS$MANAGER:DECW$STARTUP RESTART
```

On workstation systems, Digital recommends that you set DECW$CONSOLE_SELECTION to WINDOW instead of ENABLE. This setting directs console output to a Console Window application (which is new in DECwindows Motif Version 1.2–3) instead of to the operator window on the graphics screen.

Note

If your workstation uses a Qvision graphics board, you *must* set DECW$CONSOLE_SELECTION to WINDOW. (See Section 2.10.1.1 for another release note pertaining to Qvision graphics boards.)

If you prefer that console broadcasts not be disabled by default on nonworkstation systems, install the following remedial kits, which ship on the OpenVMS Version 7.1 operating system CD–ROM and are also available from your Digital support representative:

- For VAX systems, VAXMOTF07_U3012
- For Alpha systems, ALPMOTF07_U3012

Note that console broadcasts will still be disabled by default on workstations.

2.9.2.4 System Files Purged During Startup

V7.1

In the U.S. version of DECwindows Motif Version 1.2–3, the following DECwindows files are purged each time DECwindows Motif starts:

SYS$LIBRARY:DECW$*.EXE
SYS$SYSTEM:DECW$SETSHODIS.EXE

This problem is corrected in Version 1.2–4. If you are running the U.S. version of DECwindows Motif Version 1.2–3, you can correct this problem by installing the following remedial kits, which ship on the OpenVMS Version 7.1 operating system CD–ROM and are also available from your Digital support representative:

- For VAX systems, VAXMOTF07_U3012
- For Alpha systems, ALPMOTF07_U3012

2.9.3 Documentation Changes and Corrections

The following note corrects a file specification in the *Getting Started With the New Desktop* manual.

2.9.3.1 *Getting Started With the New Desktop* (Alpha Only)

DECwindows Motif V1.2–4

A file specification for a command procedure in *Getting Started With the New Desktop* (part number AA-QUW1A-TE) is incorrect. The file specification appears in Section 3.4.9, paragraph 5, as follows:

"Optional DECwindows applications, such as DECwindows Notes, may not provide any information and therefore are not restarted. For such cases, there is a command procedure called *disk*$:[*user*.DT]SESSIONETC.COM that you can use to start any applications that cannot be restarted automatically. This procedure is analogous to the DECW$LOGIN.COM procedure in the traditional DECwindows environment."

The correct file specification is:

disk$:[*user*.DT.SESSIONS]SESSIONETC.COM

2.10 DECwindows X11 Display Server (Alpha Only)

This section contains release notes pertaining to the DECwindows X11 display server for OpenVMS Alpha systems.

2.10.1 Changes and Enhancements

The following note describes a change in behavior on the DECwindows X11 display server.

2.10.1.1 Ctrl/F2 Behavior Change

V7.1

Starting with OpenVMS Alpha Version 7.1, you can no longer press Ctrl/F2 to switch from windows mode to console mode if your workstation or server contains one of these Qvision graphics boards:

- PB2GA–AA (Qvision 1024E)
- PB2GA–HA (Qvision 1280P)

This behavior has always been true for the S3 Trio64 graphics board (PB2GA–JA/JB).

Support for the operator console is now provided using a Motif-based window option that is enabled during DECwindows startup. For details, see Section 2.9.2.3.

Note that you can still use Ctrl/F2 to switch from console mode to windows mode.

2.10.2 Problems and Restrictions

The following note describes a restriction for the DECwindows X11 display server.

2.10.2.1 Graphics Boards Support for Release 6

V7.1

You must install Digital Open3D Version 4.1 for OpenVMS Alpha Version 7.1 in order to support the following types of graphics boards:

- ZLX–M
- ZLX–L
- ZLXp–L

The Digital Open3D product also provides the following 3D extensions: OpenGL, PEX, and PCM.

2.11 Digital Distributed Computing Environment (DCE) for OpenVMS

This section contains release notes pertaining to the layered product, Digital Distributed Computing Environment (DCE) for OpenVMS VAX and OpenVMS Alpha.

2.11.1 Problems and Restrictions

The following sections describe problems and restrictions associated with DCE.

2.11.1.1 DCE and Digital TCP/IP Services for OpenVMS Version 4.1

V7.1

Attempts to start DCE on OpenVMS systems running Digital TCP/IP Services for OpenVMS Version 4.1 can result in system crashes. Before you start DCE on a system running TCP/IP, install the TCP/IP mandatory update that is included on the OpenVMS VAX and OpenVMS Alpha Version 7.1 operating system CD–ROMs.

Caution

Do *not* start DCE on an OpenVMS Version 7.1 system running Digital TCP/IP Services for OpenVMS until you install the mandatory update for Version 4.1 of Digital TCP/IP Services for OpenVMS.

2.11.1.2 Support Restrictions

V7.1

To use Digital DCE for OpenVMS with OpenVMS Version 7.1, you must upgrade Digital DCE for OpenVMS to Version 1.4.

On Alpha systems, Digital DCE for OpenVMS Version 1.4 does not support multiple kernel threads.

2.12 POSIX for OpenVMS

This section contains release notes about POSIX for OpenVMS.

2.12.1 Problems and Restrictions

The following note describes a support restriction for POSIX for OpenVMS.

2.12.1.1 POSIX for OpenVMS Version 2.0 Is Not Supported

V7.0

POSIX for OpenVMS Version 2.0 is not supported on OpenVMS VAX and OpenVMS Alpha Version 7.0 and later. To use the POSIX interface on OpenVMS Version 7.0 or later, you must install POSIX for OpenVMS Version 3.0.

Caution

Do *not* try to start up POSIX for OpenVMS Version 2.0 on an OpenVMS VAX Version 7.0 or later system. Your system will crash.

2.13 Wind/U Products for OpenVMS

V7.1

Bristol Technology's Wind/U products are used by independent software vendors (ISVs) and developers to create Windows-based applications that can be deployed across multiple computing environments.

Bristol Technology's Wind/U Run-Time Library for OpenVMS is distributed with the OpenVMS Version 7.1 kit. However, you must directly contact Bristol Technology for the Wind/U Developer's Kit as well as for licenses and support for both the Run-Time Library and the Developer's Kit.

To obtain your product authorization key (PAK) to access Wind/U for OpenVMS, contact Bristol Technology as follows:

Bristol Technology
241 Ethan Allen Highway, Ridgefield, CT 06877 USA
(203) 438-6969
email: info@bristol.com

For more information about Wind/U for OpenVMS, access the Bristol Technology web site:

`http://www.bristol.com.`

3

General User Release Notes

This chapter provides information for all users of the OpenVMS operating system. It includes information about commonly used commands and utilities.

For information about new features included in this version of the software, refer to the *OpenVMS Version 7.1 New Features Manual*.

3.1 DCL Commands

This section contains release notes related to the DIGITAL Command Language (DCL) for this release of the OpenVMS operating system.

3.1.1 Changes and Enhancements

See Section 4.18.1.2 for a change in output displayed by the DIRECTORY /SECURITY and DIRECTORY/FULL commands.

3.1.2 Problems and Restrictions

The note in this section describes a restriction pertaining to DCL commands.

3.1.2.1 SET PROCESS/NOAUTO_UNSHELVE Command in Cluster Environment

V6.1

The DCL command SET PROCESS/NOAUTO_UNSHELVE does not support operations across the cluster. It can be issued only for a process on the same node, including as the default case, the process from which the command is issued.

The /IDENTIFICATION=pid qualifier is supported, but only when the target process is on the same node as the process where the command is issued.

3.1.3 Documentation Changes and Corrections

This section includes a correction to documentation of a DCL command.

3.1.3.1 SUBMIT/REMOTE Command Disallows /RETAIN Qualifier

V7.1

In the SUBMIT command documentation in the *OpenVMS DCL Dictionary* and online help, the description of the /REMOTE qualifier incorrectly lists /RETAIN as a qualifier that can be specified in a SUBMIT/REMOTE command. /RETAIN should be deleted from the list of legal qualifiers included in the /REMOTE qualifier description.

3.2 DECTPU

This section contains release notes pertaining to the DEC Text Processing Utility (DECTPU).

3.2.1 Problems and Restrictions

The note in this section describes a DECTPU problem.

3.2.1.1 Motif Widget Context Help Built-In

V1.0

The following built-in, which should enter Motif context-sensitive help mode, is disabled because of a problem in the Motif toolkit:

```
SET (WIDGET_CONTEXT_HELP, widget_variable, {on|1|off|0})
```

The mouse pointer changes to a question mark, and DECTPU waits for you to select a widget by clicking MB1. DECTPU then executes the help callback of the selected widget (or of its parent if the selected widget has no help callback). The widget_variable is the widget within which the modal help interaction will occur, usually the top-level widget returned from the GET_INFO (SCREEN, "widget") built-in. The last parameter confines the question mark pointer to the specified widget if ON or 1, and does not confine the pointer if OFF or 0.

3.3 High-Performance Sort/Merge Utility (Alpha Only)

This section contains release notes pertaining to the command line interface and the callable interface (SOR routines) of the OpenVMS Alpha high-performance Sort/Merge utility. This information is of interest to both general users and programmers.

For more information about using the OpenVMS Alpha high-performance Sort /Merge utility, refer to the *OpenVMS User's Manual* and the *OpenVMS Utility Routines Manual*.

3.3.1 Problems and Restrictions

This section describes problems associated with using the command line interface and the SOR routines of the high-performance Sort/Merge utility.

3.3.1.1 Secondary Error Messages Not Displayed

V7.0

Unlike the Sort/Merge utility, error messages generated by the high-performance Sort/Merge utility do not include secondary error messages from RMS or other facilities. For example, the Sort/Merge utility displays a secondary RMS message with the following SORT message:

```
%SORT-F-OPENOUT, error opening EXAMPLE.DAT as output
-RMS-E-FLK, file currently locked by another user
```

Under the same conditions, the high-performance Sort/Merge utility displays only the SORT message.

This difference in behavior will be fixed in a future release.

3.3.1.2 Concurrent Sort Operations

V7.0

Memory allocation differences may limit the high-performance Sort/Merge utility's ability to perform the same number of concurrent sort operations as the Sort /Merge utility can perform in the same amount of virtual memory.

If this situation occurs, you can either increase the amount of virtual memory that is available to the process, or reduce the working set extent.

3.3.1.3 Merging Stream Files Limitation

V7.0

When the high-performance Sort/Merge utility is used to merge stream files, the end-of-file is not written correctly for the output stream file. To work around this limitation, explicitly specify the format of the output file as fixed or variable:

- To do this on a MERGE command, specify the qualifier /FORMAT=FIXED:n or /FORMAT=VARIABLE:n on the output file.
- To do this at an application programming interface, specify the record format of the output file as FAB$C_FIX or FAB$C_VAR on SOR$PASS_FILES( ).

If you want the output file in stream format, use the RMS Convert utility to convert the output file from fixed or variable record format to stream format.

This limitation will be removed in a future release.

3.3.2 Corrections

This section describes problems that have been fixed in the command line interface and the SOR routines of the high-performance Sort/Merge utility in OpenVMS Version 7.1.

3.3.2.1 Default File Specification

V7.1

In OpenVMS Version 7.0, the high-performance Sort/Merge utility required the user to explicitly specify the output file extension. This behavior differed from the Sort/Merge utility, which takes the default output file extension from the first input file (when there is a user-specified input file).

This problem has been corrected in OpenVMS Version 7.1; you no longer need to explicitly specify the output file extension when using the high-performance Sort/Merge utility.

3.4 Online Help

This section contains release notes pertaining to online help. Also see Section 3.1.3.1 for a correction to online help.

3.4.1 Changes and Enhancements

This section describes changes to online help.

3.4.1.1 Topic Name Changes

V7.1

The titles of several help topics have been changed to promote usability. The content of these help topics is comparable to the content in previous releases. Changes affect the following topics:

Old Topic Title	New Topic Title	Contents
System_Files	Sys_Files	Describes content of system files.
Specify	DCL_Tips	Describes syntax conventions for DCL commands.

3.5 Point-to-Point Protocol Utility (Alpha Only)

This section contains release notes pertaining to the Point-to-Point Protocol utility (PPPD) on OpenVMS Alpha Version 7.1 systems.

3.5.1 Problems and Restrictions

The following note outlines TCP/IP requirements for using the Point-to-Point Protocol utility on OpenVMS Alpha Version 7.1 systems.

3.5.1.1 Internet Protocol Callback Images Requirement

V7.1

The Point-to-Point Protocol utility (PPPD) included with OpenVMS Alpha Version 7.1 provides point-to-point networking capability to the base operating system. However, for this utility to be fully functional, your current TCP/IP vendor must also develop and integrate the appropriate callback images into their Internet Protocol (IP) stack.

The Digital TCP/IP Services for OpenVMS team is currently integrating these images into the next version of their product. Digital is also working in tandem with our third-party TCP/IP vendors to provide this functionality; contact your individual TCP/IP vendor for more details.

If you receive the following error message while using PPPD, contact your system manager to verify whether PPPD is currently functional on your OpenVMS system.

```
%PPPD-E-NOTREG, network protocol has not been registered
```

For more information about the callback images, refer to the following files:

- SYS$SYSROOT:[SYSHLP.EXAMPLES.PPPD.DOC]PPP_INTERFACES.*

  Provides an overview of the callback images.

- SYS$SYSROOT:[SYSHLP.EXAMPLES.PPPD]SAMPLE_PPP_CALLBACK_SETUP.TXT

  Describes the sample build procedures, provided by Digital, that IP developers can use to construct and test their callback images.

For information about PPPD, see the *TCP/IP Networking on OpenVMS Systems* manual.

4

System Management Release Notes

This chapter contains information that applies to system maintenance and management, performance management, and networking.

For information about new features included in this version of the software, refer to the *OpenVMS Version 7.1 New Features Manual*.

4.1 Alpha System Dump Analyzer (SDA)

The following release notes pertain to the Alpha System Dump Analyzer (SDA).

4.1.1 Corrections

The following note describes a correction to Alpha SDA.

4.1.1.1 SHOW POOL/RING_BUFFER Command Now Displays Large Pool Packets

V7.1

The SHOW POOL/RING_BUFFER command now correctly displays pool packets whose size exceeds 65535 bytes.

For more detailed information, see the *OpenVMS Alpha System Dump Analyzer Utility Manual*.

4.2 AUTOGEN Command Procedure

This section contains release notes pertaining to the AUTOGEN command procedure.

4.2.1 Changes and Enhancements

This section describes changes and enhancements to AUTOGEN on OpenVMS systems.

4.2.1.1 NPAGEDYN and NPAGEVIR Limitations and Warnings (VAX Only)

V7.0

For the benefit of OpenVMS VAX systems with limited physical memory, AUTOGEN logs a warning message in its report if NPAGEDYN exceeds 10 percent of physical memory or if NPAGEVIR exceeds 33 percent of physical memory.

AUTOGEN also limits its own calculated value for NPAGEDYN to 20 percent of physical memory, and limits NPAGEVIR to 50 percent of physical memory. These calculated values are adequate for most workstations and systems with 16 or fewer megabytes of physical memory. If your system requires a larger value, you can override the AUTOGEN calculated values by setting higher values in MODPARAMS.DAT.

4.2.1.2 Calculating the Page File Size (Alpha Only)

V7.0

On OpenVMS Alpha systems, the formula for calculating the size of page file space has changed. For details about calculating the size of page file space, refer to the *OpenVMS System Manager's Manual*.

4.2.1.3 WSMAX Calculations

V7.0

Starting with OpenVMS Alpha Version 7.0 and OpenVMS VAX Version 6.2, the calculation of WSMAX is no longer linear but resembles a logarithmic curve. Instead of calculating WSMAX as being a quarter of the size of physical memory, it is now calculated as a quarter of the first 32 MB, plus a sixteenth of the memory from 32 to 256 MB, plus a sixty-fourth of the memory (if any) above 256 MB.

This is intended to assist managers of systems that host large numbers of users whose working sets are not large. Systems whose user bases consist of a small number of users (or processes) that require large amounts of physical memory (for example, simulations) might need to set MIN_WSMAX to a value that satisfies the requirements of those processes.

4.3 Backup Utility

Release notes in this section pertain to the Backup utility (BACKUP).

4.3.1 Changes and Enhancements

The note in this section describes a change in behavior.

4.3.1.1 /SINCE Qualifier Behavior Change

V7.1

With OpenVMS Version 6.2, the Backup utility was changed so that specifying the /SINCE qualifier caused more files to be saved than in previous releases. This change was made to ensure the integrity of the contents of incremental save sets.

For users who prefer the pre-Version 6.2 behavior when performing Backup operations that use the /SINCE qualifier, Version 7.1 includes a workaround, as shown in the following example:

```
$ CREATE/NAME_TABLE BACKUP$BTE
$ DEFINE/TABLE=BACKUP$BTE BACKUP$BTE_DISABLE_SAVE_ALL_DIR " "
```

The first command creates a table called BACKUP$BTE. The second command inserts the logical name BACKUP$BTE_DISABLE_SAVE_ALL_DIR into the table. This logical invokes the pre-Version 6.2 behavior for all Backup sessions in that process that use the /SINCE qualifier on any operation.

This temporary workaround will be replaced by a new qualifier in a future release.

4.3.2 Problems and Restrictions

This section describes known problems and restrictions for the Backup utility.

4.3.2.1 /IMAGE and /ALIAS Qualifiers

V7.1

Specifying the /IMAGE qualifier without also specifying /NOALIAS can result in incomplete disk or file restoration operations. Therefore, Digital strongly recommends that you specify /NOALIAS with /IMAGE when performing image mode backup operations.

Note

If you do not specify /NOALIAS, the /ALIAS qualifier is activated by default.

When a save set is created using /IMAGE and /ALIAS (explicitly or by default) in OpenVMS Versions 6.2 and 7.0, Backup saves only one copy of a file: either the alias file entry or the primary file entry. If the primary file entry is not saved in the save set, subsequent restore operations for this save set would restore the file using its alias entry, causing the file header of the created file to contain the wrong file name.

If you use /NOALIAS to restore a volume from a save set created using /ALIAS in Version 6.2 or 7.0, the volume might be incompletely restored. If Backup previously saved alias file entries instead of primary file entries, the alias file entries would be omitted from a volume restored using /NOALIAS.

To safely restore a save set created using /ALIAS in Version 6.2 or 7.0, use the following procedure:

1. Restore the save set using the /IMAGE and /ALIAS qualifiers.
2. Correct file entries by renaming the offending files, as shown in the following example:

```
$ RENAME DISK:[000000]VMS$COMMON.DIR DISK:[000000]SYSCOMMON.DIR
$ RENAME DISK:[000000]SYSCOMMON.DIR DISK:[000000]VMS$COMMON.DIR
```

   In this example, VMS$COMMON.DIR is the primary file entry and SYSCOMMON.DIR is the alias file entry.

Starting with OpenVMS Version 7.1, if you specify /IMAGE without /NOALIAS, Backup saves both the primary and alias file entries.

4.3.2.2 /VERIFY Qualifier

V7.1

Disk-to-disk copy operations initiated using the /VERIFY qualifier may attempt to verify files that are not copied. For example, if an error prevents you from successfully copying a file from one disk to another location and you specified the /VERIFY qualifier for that operation, the system displays two error messages: one indicating that the file was not copied and another indicating that the file was not verified.

4.3.2.3 /ENCRYPTION Qualifier Not Supported (Alpha Only)

V7.1

Use of the /ENCRYPTION qualifier is not supported on OpenVMS Alpha systems. This problem will be addressed in a future release.

4.3.2.4 MOUNT Messages When Backing Up Tapes

V7.1

The MOUNT utility generates VOLINV messages on continuation tape volumes during backups when you use devices that have loaders or when the stackers or loaders become empty. Following is an example of the messages displayed during a BACKUP/NOASSIST operation:

```
%MOUNT-I-MOUNTED, ABCD03 mounted on _$4$MUA3: (HSC70)
%BACKUP-I-RESUME, resuming operation on volume 4
%MOUNT-F-VOLINV, volume is not software enabled
%BACKUP-I-READYWRITE, mount volume 4 on _$4$MUA3: for writing
Enter "YES" when ready: yes
%MOUNT-I-MOUNTED, ABCD04 mounted on _$4$MUA3: (HSC70)
```

Once the devices are put back on line or the media is made ready, the backup session continues or finishes as expected. This problem will be addressed in a future release.

4.3.2.5 FILES–11 Mounted Tapes

V7.1

When tapes are mounted in FILES–11 mode, Backup outputs an ACCVIO message on a write operation or reports "file not found" on a READ operation instead of reporting a fatal error message stating that a FOREIGN mounted tape volume is required. Following is an example of the messages displayed during a BACKUP/LIST operation:

```
%BACKUP-F-OPENIN, error opening $4$MUA3:[TEST].; as input
-RMS-E-FNF, file not found
```

This problem will be addressed in a future release.

4.3.2.6 Image Backups from an RF73 Disk

V6.1

When performing an image backup from an RF73 disk (or a disk with a cluster size of 4 blocks) to an RF74 disk (or a disk with a cluster size of 7 blocks), the Backup utility does not check the file size when it is allocating space for the file being copied. Therefore, if the file has an allocation greater than the value of the CLUSTER_SIZE attribute established during initialization, the Backup utility will allocate one more cluster size number of blocks to the allocation size even though the actual file size is less than the cluster size. For example, during an image backup, a file that uses 6 blocks and is allocated 8 blocks (which displays as 6/8 on the screen if you enter a DIRECTORY/SIZE=ALL command) shows an increase in its allocation size to 14, instead of 7, after it is copied to the target disk.

As a result of this problem, the following files are copied to the image system disk with a blocks used/allocation size of 6/14 blocks:

SYS$COMMON:[SYS$LDR]LIDRIVER.EXE
SYS$COMMON:[SYS$LDR]LPDRIVER.EXE

This incorrect allocation size causes standalone BACKUP to *fail* on the booted image system disk.

To correct this problem, recopy the two previously listed files to the same directory after the image backup, by using the following command (which also specifies the correct allocation size):

```
$ COPY/ALLOCATION=7 SYS$COMMON:[SYS$LDR]LIDRIVER.EXE SYS$COMMON:[SYS$LDR]
$ COPY/ALLOCATION=7 SYS$COMMON:[SYS$LDR]LPDRIVER.EXE SYS$COMMON:[SYS$LDR]
```

4.4 Debugger

This section describes system management release notes for the OpenVMS Debugger. Debugger programming release notes are in Section 5.3.

4.4.1 Problems and Restrictions

The following note describes a restriction when using the OpenVMS Debugger.

4.4.1.1 Displaying a Debug Session from a Personal Computer (PC)

V7.0

Although displaying a debug session from a PC is not officially supported or tested at this time, many users have reported successful results when using the following configurations:

- OpenVMS Debugger Version 6.2 or greater (earlier versions had some geometry problems)
- X-Windows Emulators:
  - Under Microsoft Windows: eXcursion Version 1.2A-1 (Win16)
  - Under MS–DOS: PC DECwindows Motif Version 5.1.005 (also known as DWDOS)
- Transports:
  - DEC TCP/IP Services for OpenVMS using UCX Version 3.1 or 3.2
  - PATHWORKS Version 4.1 or 4.2
  - Windows Sockets TCP/IP Version 1.1

Several combinations of VAXstations, AlphaStations, 486, and Pentium processors have been tried, with no serious restrictions or performance problems.

Following are some general tips:

- Use a high-resolution monitor. A typical 640-X-480 pixel VGA display is too small. You can fine-tune the geometry and placement of your OpenVMS Debugger windows by editing your VMSDEBUG.DAT file (using the customization features described in the *OpenVMS Debugger Manual*). It is suggested that you make the windows as small as you are comfortable with, and overlap their placement as needed to minimize display space requirements.
- Users of DECterm emulators should restrict their use to the screen-mode command interface, and not try to run the DECwindows Motif interface. Screen-mode debugging should be successful with any reasonably X-compliant terminal emulator, such as a DECterm (that is, created with the CREATE /TERM/DETACH command).

4.5 DECamds

This section contains release notes pertaining to DECamds. DECamds is a separately installable, real-time, high-performance, multisystem monitoring utility that is operated from a centralized, mouse-driven DECwindows display. For more information about DECamds, refer to the *DECamds User's Guide*, which ships on the OpenVMS Version 7.1 Documentation CD–ROM.

4.5.1 Changes and Enhancements

The following section describes a change in licensing for DECamds.

4.5.1.1 License is Now Included with OpenVMS

V7.1

Starting with OpenVMS Version 7.1, the right to use DECamds will be provided with the OpenVMS base operating system license. Prior to Version 7.1, the right to use DECamds was provided with the OpenVMS Cluster software license. This change was made in response to customer demand to use the system management capabilities provided by DECamds in a nonclustered environment.

4.5.2 Problems and Restrictions

The following section describes a DECamds restriction.

4.5.2.1 Kernel Threads Not Supported (Alpha Only)

V7.0

DECamds support for kernel threads has not been implemented on OpenVMS Alpha systems. If you use threaded processes, DECamds displays only the top thread.

4.6 DECdtm Services

This section contains release notes pertaining to DECdtm services.

4.6.1 Problems and Restrictions

This section describes known problems and restrictions associated with using DECdtm services.

4.6.1.1 Kernel Threads Restriction (Alpha Only)

V7.1

On OpenVMS Alpha systems, unpredictable results can occur if DECdtm services are used in a multithreaded environment. Do not make calls to DECdtm services in kernel threads other than the root thread because much of the work performed by DECdtm uses the context of the calling process.

4.7 DECevent Fault Management Support (Alpha Only)

Alpha V6.2

Starting with OpenVMS Alpha Version 6.2, DECevent replaces the Error Reporting Formatter (ERF) as the bit-to-text translating tool for fault management on OpenVMS systems. While the ERF is still available, it will be retired in a future release of the OpenVMS operating system.

Refer to Section 1.4.1.1 for information about a change in how the DECevent DIAGNOSE command is enabled starting with OpenVMS Alpha Version 7.1.

4.7.1 Problems and Restrictions

This section describes problems and restrictions related to using the DECevent fault management tool.

4.7.1.1 Bit-to-Text Translation Support

V6.2

DECevent bit-to-text translation is supported on many products and devices. On other devices, as much translation as possible is performed and all remaining information in the event is dumped in hexadecimal.

Contact your Digital support representative if you have questions about the type of DECevent support currently available for your devices.

4.7.1.2 Logical File Names

V6.2

DECevent is unable to translate as input any logical defined as a search list of file names. For example:

```
$ DEFINE EVENT_LOG DISK1:[EVENTS]EVENT_LOG1.SYS,DISK1:[EVENTS]EVENT_LOG2.SYS
$ DIAGNOSE/ANALYZE EVENT_LOG

DECevent T1.0 FT2
_DIAGNOSE-FAT: Analyze - No files found ' event_log '
_DIAGNOSE-FAT: An error occurred while executing a command ruleset
_DIAGNOSE-INF: No Error Messages to send in thread 1
```

4.8 External Authentication

This section contains release notes pertaining to external authentication. External authentication is an optional feature introduced in OpenVMS Version 7.1 that enables OpenVMS systems to authenticate designated users using their LAN Manager user IDs and passwords.

To enable external authentication on your system, the following are required:

- PATHWORKS Version 5.0E for OpenVMS, operating as a LAN Manager domain member, BDC, or PDC
- Windows NT server (if used) must enable LAN Manager Version 2.x updates
- DECwindows Version 1.2–4

For more information about external authentication, refer to the *OpenVMS Version 7.1 New Features Manual* or the version of the *OpenVMS Guide to System Security* that ships on the OpenVMS Version 7.1 Documentation CD–ROM.

4.8.1 Changes and Enhancements

The following note describes a change in behavior related to external authentication.

4.8.1.1 OpenVMS User Name Prompt Accepts Case-Sensitive Terminal Input

V7.1

You can enter a case-sensitive user name at the OpenVMS user name prompt if you enclose it in quotation marks. If you do not enclose the user name in quotation marks, LOGINOUT converts the user name to uppercase characters.

OpenVMS and LAN Manager user names are not case-sensitive. Therefore, quotation marks are not necessary if you enter an OpenVMS user name or a LAN Manager user ID. In future releases, external authentication may support other authentication services that allow case-sensitive user names.

You can restore previous behavior on your OpenVMS system by setting the forced uppercase configuration bit in the SYS$SINGLE_SIGNON logical name. (See the *OpenVMS Version 7.1 New Features Manual* or the Managing System Access chapter in the *OpenVMS Guide to System Security* for more information.)

4.8.2 Problems and Restrictions

The following notes describe problems or restrictions related to external authentication.

4.8.2.1 Impact on Layered Products and Applications

V7.1

Certain layered products and applications that use an authentication mechanism based on the traditional SYSUAF-based user name and password (for example, software that calls $HASH_PASSWORD or $GETUAI/$SETUAI to alter, fetch, or verify OpenVMS passwords) will encounter problems in either of the following cases:

- When external authentication is used in an environment where a given user's LAN Manager user ID and OpenVMS user name are different
- Where the user's SYSUAF password is different than the LAN Manager password

In such cases, the problem symptom is a user authentication failure from the layered product or application.

For externally authenticated users, the normal system authorization database (SYSUAF.DAT) is used to construct the OpenVMS process profile (UIC, privileges, quotas, and so on) and to apply specific login restrictions. However, there are two key differences between externally authenticated users and normal OpenVMS users. For externally authenticated users:

- The password stored in the SYSUAF is not the password used to verify the user
- The user name stored in the SYSUAF and used to identify the OpenVMS process is not necessarily the same as the LAN Manager user ID used to authenticate the user during login

OpenVMS attempts to keep a user's SYSUAF and LAN Manager password synchronized in order to minimize these problems. An up-to-date copy of the user's LAN Manager password is kept in the SYSUAF, but this is not the case if, for example, the LAN Manager password contains characters that are invalid in OpenVMS, or SYSUAF password synchronization is disabled by the system manager. (Password synchronization is enabled by default.)

If you enable external authentication, Digital recommends you do the following to minimize incompatibility with layered products or applications that use traditional SYSUAF-based authentication:

- Do not disable password synchronization.
- Limit LAN Manager passwords to those characters from the OpenVMS valid password character set (A–Z, 0–9, underscore (_), and dollar sign ($)).
- Assign users the same user name in both LAN Manager and OpenVMS.
- Do not assign the same user name or user ID to more than one user.

The $GETUAI and $SETUAI system services do not support external passwords. These services operate only on passwords stored in the SYSUAF and updates are not sent to LAN Manager. Enabling external authentication is not recommended for sites using software that makes calls to these services for the purposes of password checking or updates. Digital expects to provide a new programming interface for this purpose in a future release.

4.8.2.2 Mixed-Version OpenVMS Cluster Systems

V7.1

Digital recommends using external authentication on OpenVMS Cluster systems only if all systems are running OpenVMS Version 7.1.

When external authentication is enabled on systems running versions of OpenVMS earlier than Version 7.1, only the Version 7.1 systems directly interact with LAN Manager. The earlier version systems continue to use the SYSUAF file for authentication and management of passwords.

If password synchronization is enabled on the Version 7.1 systems (the default), the SYSUAF passwords are kept synchronized with LAN Manager passwords and users can log in to the earlier version systems using their OpenVMS user names and passwords. LAN Manager user IDs cannot be used on the earlier version systems. (If a site maintains identical OpenVMS user names and LAN Manager user IDs, this is not an issue.)

LOGINOUT on earlier version systems continues to enforce normal OpenVMS password policy (password expiration, password history, and so on), on all users, including externally authenticated users.

4.8.2.3 LGI Callout Services Disable External Authentication

V7.1

In Version 7.1, the presence of LOGINOUT (LGI) callouts disables external authentication. This restriction is expected to be removed in a future release.

4.8.2.4 DECwindows Pause Screen Uses SYSUAF Password

V7.1

The DECwindows pause screen unlock mechanism does not use LAN Manager for password validation. It continues to use the password in the SYSUAF file, even if you have external authentication enabled on your system.

Password synchronization is enabled by default. If you have disabled password synchronization, be sure to keep the LAN Manager and SYSUAF passwords synchronized manually.

4.8.2.5 FTP Server and Failed Connections With External Authentication

V7.1

The File Transfer Protocol (FTP) server does not use external authentication to authenticate FTP connections on the OpenVMS system. This causes connects to fail if either of the following conditions exist:

- The LAN Manager user ID is different from the OpenVMS user name.
- The OpenVMS password is not synchronized with the LAN Manager password.

4.8.2.6 DECnet-Plus and NET_CALLOUTS Parameter

V7.1

In order to run DECnet-Plus for OpenVMS with external authentication enabled, set the SYSGEN parameter NET_CALLOUTS to 255. This enables LAN Manager user ID mapping and authentication for network logins.

4.8.2.7 PATHWORKS LAN Manager Server Required for External Password Updates

V7.1

External authentication using LAN Manager requires a running LAN Manager server to support external password changes initiated by users. Password changes fail if a LAN Manager server is not present.

This restriction will be removed in a future release.

4.8.2.8 No Password Expiration Notification on Workstations

V7.1

In the LAN Manager domain, a user cannot log in once a password expires.

Users on personal computers (PCs) receive notification of impending LAN Manager password expiration, and can change passwords before they expire. However, when a user logs in from an OpenVMS workstation using external authentication, the login process cannot determine if the external password is about to expire. Therefore, sites that enforce password expiration, and whose user population does not primarily use PCs, may elect not to use external authentication for workstation users.

This problem will be fixed in a future release.

4.9 Install Utility

This section contains release notes pertaining to the Install utility (INSTALL).

4.9.1 Changes and Enhancements

This section describes a change to INSTALL.

4.9.1.1 Installing Images

V6.2

The REPLACE option for the OpenVMS Install utility (INSTALL) has been changed to modify the known file database in an atomic fashion.

In the past, REPLACE was equivalent to DELETE followed by ADD. Consequently, there was a short time during which neither the new nor the old image was in the known file database. When activating protected or privileged images, this could result in failed image activations. Also, if the new image

could not be installed, it was possible for neither the old nor the new image to be installed after the failure.

These problems are now corrected.

With the change, REPLACE operations for images installed with the /SHARED qualifier might require more global sections or global pages than in the past. Also, the names of global sections have been changed to avoid naming conflicts. The global sections can be displayed with one of the following commands:

```
$ INSTALL LIST /GLOBAL
```

or

```
$ INSTALL LIST image-name /GLOBAL
```

4.10 LAN ATM (Alpha Only)

This section contains release notes pertaining to the LAN ATM (asynchronous transfer mode) software.

4.10.1 Problems and Restrictions

This section describes a problem with the LAN ATM Software.

4.10.1.1 ATM Switch Problem

V7.1

The LAN ATM software is set to automatically sense the User-to-Network Interface (UNI) version. Some ATM switches do not work with this feature. If your emulated LAN or classical IP subnet cannot be enabled, set both the ATM switch and the OpenVMS Alpha system to run at the same UNI version. (Use the SYSGEN parameter LAN_FLAGS to set the OpenVMS system.)

4.11 LANCP/LANACP Servers

This section contains release notes pertaining to the LANCP/LANACP servers.

4.11.1 Changes and Enhancements

The following note describes a change that affects how the LANACP LAN server application is started.

4.11.1.1 LANACP LAN Server Application Startup (Alpha Only)

V7.1

The LANACP LAN server application is now started in VMS$DEVICE_STARTUP.COM. The LAN$STARTUP command has been removed from the system startup template file, SYSTARTUP_VMS.TEMPLATE, because the command is no longer necessary.

4.12 Mail Utility

This section contains release notes about the Mail utility that are of interest to system managers. Release notes of interest to programmers are documented in Section 5.14.

4.12.1 Changes and Enhancements

The following note describes an important change to Mail.

4.12.1.1 MAIL.EXE and Privileges

V7.0

The OpenVMS installation procedure does not initially install MAIL.EXE with any privileges (because MAIL.EXE does not require privileges to perform its functions). Prior versions of the OpenVMS operating system did include mechanisms that allowed MAIL.EXE to check, ignore, grant, or override certain privileges that a system manager might assign when reinstalling MAIL.EXE. Because these regulatory mechanisms sometimes created unexpected or undesirable conditions, they have been removed in Version 7.0 of the OpenVMS operating system.

Caution

If you reinstall MAIL.EXE with certain privileges, you must carefully consider possible ramifications, including the potential for security breaches. For example, because MAIL.EXE confers its privileges on any user who invokes the Mail utility, that user will inherit those privileges if the user creates a subprocess from within Mail by specifying the SPAWN command.

4.13 Mount Utility

The following release notes pertain to the Mount utility (MOUNT).

4.13.1 Corrections

This section describes corrections to MOUNT.

4.13.1.1 MOUNT/CLUSTER and DISMOUNT/CLUSTER Commands Function Correctly

V7.1

The Mount utility has been completely rewritten for OpenVMS Version 7.1 and is now significantly faster and more robust.

In previous releases, some cluster configurations could not successfully use the DCL commands MOUNT/CLUSTER and DISMOUNT/CLUSTER, and were forced to use workarounds such as using the SYSMAN utility to issue remote node MOUNT and DISMOUNT commands. With OpenVMS Version 7.1 this restriction is removed. The MOUNT/CLUSTER and DISMOUNT/CLUSTER commands now function correctly under all circumstances.

4.14 Nonpaged Pool

This section contains release notes pertaining to nonpaged pool.

4.14.1 Changes and Enhancements

This section describes changes to nonpaged pool.

4.14.1.1 Reclamation Algorithms Changed (Alpha Only)

V7.1

On OpenVMS Alpha systems, nonpaged pool reclamation algorithms have been changed to ensure that packets put onto lookaside lists are returned to variable pool in a timely fashion. Both gentle and aggressive reclamation algorithms have been changed to return more packets to variable pool.

Reclamation of a list first puts all the packets on the list back into variable pool, and then repopulates the list to the requested percentage of original packets.

Each gentle reclamation pass processes two lists. Therefore, using the 30-second default interval, all 80 lists are processed every 20 minutes.

If you have some very long lookaside lists (because of increased activity), and the packets are no longer frequently used, take the following temporary actions:

- Decrease the value of the NPAG_GENTLE system parameter so that a smaller percentage of packets remains on the list after reclamation.
- Decrease the value of the NPAG_INTERVAL system parameter so that reclamation occurs more frequently.

Aggressive reclamation still processes lists until one of the following events occurs:

- The allocation request is satisfied.
- The request cannot be satisfied from existing pool, so pool is expanded until the request can be satisfied.

If you are willing to use extra cycles to avoid expanding pool, decrease the value of the NPAG_AGGRESSIVE system parameter. This causes more packets to return to variable pool, thus increasing the chances of satisfying a request without expanding pool.

Refer to the *OpenVMS Version 7.1 New Features Manual* for descriptions of the new system parameters that are related to changed reclamation algorithms.

4.15 OpenVMS Cluster Systems

The following sections contain release notes pertaining to OpenVMS Cluster systems.

4.15.1 Changes and Enhancements

This section contains notes about changes and enhancements to OpenVMS Cluster systems.

4.15.1.1 Cluster Client License Changes

V7.1

A change has been made to OpenVMS Cluster Client licensing. Prior to Version 7.1, the OpenVMS Cluster Client license enabled full OpenVMS Cluster functionality, with the following exceptions:

- Client CPUs could not provide votes toward the operation of the OpenVMS Cluster system.
- Client CPUs could not use MSCP to serve disks or TMSCP to serve tapes.

Previously, the first exception regarding voting was not enforced. Starting with Version 7.1, this exception is enforced.

4.15.1.2 OpenVMS Cluster Compatibility Kit for Version 6.2 Systems

V7.1

The OpenVMS Cluster Compatibility Kit provides many OpenVMS Version 7.1 enhancements for Version 6.2 systems. This kit is required for Version 6.2 systems if they are included in a cluster with Version 7.1 systems (same system architecture or a mix of VAX and Alpha systems). Optionally, users can install it on other OpenVMS Version 6.2 systems to derive the same benefits.

Cluster Compatibility Kit Features

The Cluster Compatibility Kit includes the following improvements for systems running OpenVMS Version 6.2:

- OpenVMS Version 7.1 Volume Shadowing enhancements

  The volume shadowing enhancements include significant quality improvements and an increase in supported shadow set members from 400 to 500. Note that the Version 7.1 volume shadowing system disk minimerge feature is not included in the Cluster Compatibility Kit nor is the Dump file off the system disk for OpenVMS Alpha. (Dump file off the system disk has been available for OpenVMS VAX systems since Version 6.2.)

  Note

  If you use volume shadowing, be sure to read the volume shadowing release notes in Section 4.24.

- OpenVMS Version 7.1 Mount enhancements

  The Mount utility has been completely rewritten, resulting in a faster, more robust utility.

- Correction to a lock manager problem found in OpenVMS Version 6.2

  The lock manager changes correct a problem in OpenVMS Version 6.2 that could corrupt some internal states in lock information used by fork lock routines, notably the I/O cache subsystem. This problem was corrected in OpenVMS Version 7.0, and also in the limited hardware releases for OpenVMS Alpha Version 6.2-1H2 and Version 6.2-1H3.

- Limited support for new port allocation classes for SCSI devices

  Port allocation classes are a new naming option for SCSI devices on systems running OpenVMS Alpha Version 7.1. If you have installed the Cluster Compatibility Kit (on a VAX or Alpha system), you can access SCSI disks on an OpenVMS Alpha Version 7.1 system that use port allocation classes in their names, but you cannot name SCSI disks on a Version 6.2 system with port allocation classes.

Snapshot Facility Disabled

Installing the Cluster Compatibility Kit on a Version 6.2 system disables the Snapshot facility, which has been removed from OpenVMS VAX Version 7.1 (see Section C.9).

System Dump Analyzer Utility (SDA)

A special version of the OpenVMS Version 6.2 System Dump Analyzer (SDA) utility is included in the Cluster Compatibility Kit. It recognizes the new Volume Shadowing data structures.

When you install the Cluster Compatibility Kit, the existing OpenVMS Version 6.2 SDA is renamed SDA_OLD.EXE and the Cluster Compatibility Kit version is named SDA.EXE. Use SDA_OLD.EXE to analyze crash dumps from an OpenVMS Version 6.2 system that has not installed the Cluster Compatibility Kit. Use SDA.EXE to analyze crash dumps from an OpenVMS Version 6.2 system that has installed the Cluster Compatibility Kit.

Installing the Cluster Compatibility Kit

Two kits are available, one for OpenVMS VAX Version 6.2 systems and one for OpenVMS Alpha Version 6.2 systems. Each kit is on its respective OpenVMS CD–ROM, as follows:

- [CLUSTER_COMP_VAX062]VAXCOMPAT_062.A (for VAX)
- [CLUSTER_COMP_ALPHA062]ALPCOMPAT_062.A (for Alpha)

To install the VAX kit, use the following command:

```
$ @SYS$UPDATE:VMSINSTALL VAXCOMPAT_062.A ddcu:[CLUSTER_COMP_VAX062]
```

To install the Alpha kit, use the following command:

```
$ @SYS$UPDATE:VMSINSTALL ALPCOMPAT_062.A ddcu:[CLUSTER_COMP_ALPHA062]
```

4.15.1.3 KFPSA Adapter Support (Alpha Only)

V7.1

The OpenVMS Alpha Version 7.1 operating system supports the KFPSA, a PCI-to-DSSI adapter, on all PCI-based AlphaServer computers.

Previously, a maximum of four KFPSA adapters was allowed per system. With OpenVMS Version 7.1, the maximum has been raised to 24.

4.15.1.4 Maximizing CIPCA Adapter Performance With an HSJ50

V7.1

To maximize the performance of the CIPCA adapter with an HSJ50 controller, Digital recommends that you enable the use of 4K CI packets by the HSJ50. To do this, your HSJ50 firmware revision level must be at Version 5.0J–2 or higher (see Section 4.15.2.4.2).

To enable the use of 4K CI packets, specify the following command at the HSJ50 console prompt:

```
CLI> SET THIS_CONTROLLER CI_4K_PACKET_CAPABILITY
```

This command takes effect when the HSJ50 is rebooted.

4.15.1.5 Fast Path Support for CIPCA

Fast Path, introduced in OpenVMS Version 7.0, is an optional, high-performance feature designed to improve I/O performance. In OpenVMS Version 7.1, Fast Path support for disk I/O has been extended to the CIPCA port. For OpenVMS Version 7.0, support was limited to the CIXCD port.

CI Fast Path disk I/O is now supported for both PCI and XMI based systems. For more information about Fast Path, see the *OpenVMS I/O User's Reference Manual*.

4.15.1.6 CIPCA Use of Synchronous Arbitration Algorithm

CIPCA uses a new, optimal CI arbitration algorithm, called Synchronous Arbitration, instead of the older Asynchronous Arbitration algorithm. The two algorithms are completely compatible. Under CI saturation conditions, both the old and new algorithms are equivalent and provide equitable round-robin access to all nodes. However, under less than saturation conditions, the new algorithm provides the following benefits:

- Reduced packet transmission latency due to reduced average CI arbitration time.
- Increased node-to-node throughput.
- Complete elimination of CI collisions that waste bandwidth and increase latency in configurations containing only Synchronous Arbitration nodes.
- Complete compatibility on the same CI between nodes using Synchronous Arbitration and nodes using the older Asynchronous Arbitration (with CIXCDs, HSCs, CIBCAs, and so forth).
- Reduced CI collision rate in configurations with mixed Synchronous and Asynchronous Arbitration CI nodes. The reduction is roughly proportional to the fraction of CI packets being sent by the Synchronous Arbitration CI nodes.

Support for Synchronous Arbitration is latent in the HSJ controller family. In configurations containing both CIPCAs and HSJ controllers, Digital recommends enabling the HSJs to use Synchronous Arbitration. Use the following HSJ CLI command to do this:

```
CLI> SET THIS CI_ARB = SYNC
```

This command takes effect upon the next reboot of the HSJ.

4.15.2 Problems and Restrictions

This section describes problems and restrictions pertaining to OpenVMS Cluster systems.

4.15.2.1 Booting Satellites with DECnet–Plus

V7.1

Digital recommends the use of the LANCP utility for all MOP booting requirements. If you choose to use DECnet-Plus MOP booting instead of LANCP, note the following restriction when adding a satellite: CLUSTER_CONFIG.COM uses the first circuit configured for MOP in the NET$MOP_CIRCUIT_STARTUP.NCL file.

To use a different circuit, you must edit NET$MOP_CIRCUIT_STARTUP.NCL before invoking CLUSTER_CONFIG.COM. Place your desired circuit at the beginning of the NET$MOP_CIRCUIT_STARTUP.NCL file; then invoke CLUSTER_CONFIG.COM.

4.15.2.2 SCSI Cluster Restrictions (Alpha Only)

V7.1

Most SCSI cluster restrictions are described in the SCSI OpenVMS Cluster appendix in *Guidelines for OpenVMS Cluster Configurations*. (See Section 4.15.3.1 for a correction to that appendix.) Section 4.15.2.3 describes a SCSI cluster restriction that is *not* documented in the manual.

4.15.2.3 AlphaServer 4000/4100 Systems Problem in SCSI Clusters

V7.1

An AlphaServer 4000/4100 system that accesses its system disk through a KZPSA adapter may not boot or write a crash dump file if another system on the SCSI bus is booting or shutting down at the same time. Subsequent attempts to boot should succeed.

The system firmware on the Version 3.8 firmware CD–ROM, and earlier versions, exhibit this problem. A correction to this problem is planned for the next firmware CD–ROM release, Version 3.9.

Digital recommends that you not attempt to perform these operations simultaneously in this configuration until you have updated your firmware.

4.15.2.4 CI-to-PCI (CIPCA) Adapter (Alpha Only)

V7.1

The release notes in this section describe restrictions for using the CIPCA module on OpenVMS Alpha systems.

4.15.2.4.1 HSC40 and HSC70 Controller Revision Levels

The HSC40 and HSC70 controllers must have a Revision F (or higher) L0109 module. If your HSC40 or HSC70 controller does not meet this requirement, contact your Digital support representative.

4.15.2.4.2 HSJ50 Firmware Requirement for Use of 4K CI Packets

Do not attempt to enable the use of 4K CI packets by the HSJ50 controller unless the HSJ50 firmware is Version 5.0J–2 or higher. If the HSJ50 firmware version is less than Version 5.0J–2 and 4K CI packets are enabled, data can become corrupted. If your HSJ50 firmware does not meet this requirement, contact your Digital support representative.

For more information about the use of 4K CI packets by the HSJ50 controller, see Section 4.15.1.4.

4.15.2.4.3 Link Module DIP Switch Settings

For link module Rev. A01, use the DIP switch settings described here to prevent arbitration timeout errors. Under heavy CI loads, arbitration timeout errors can cause CI path errors and CI virtual circuit closures.

The DIP switch settings on the CIPCA link module select cluster size and node address. Follow these instructions when setting the DIP switches for link module Rev. A01 only:

- If the cluster size is set to 16, do not set a CI adapter to node address 15 on that star coupler.
- If the cluster size is set to 32, do not set a CI adapter to node address 31 on that star coupler. Also, do not set any CIPCA to node address 0 *or* do not set any CI adapter to node address 16.

These restrictions do not apply to link module Rev. B01 and higher.

4.15.2.4.4 Systems Using CIPCA, CIXCD, and KFMSB

The following restrictions apply when CIPCA, CIXCD, and KFMSB adapters are used in systems:

- AUTOCONFIGURE must be run before the end of booting.

  When you perform a normal installation boot, AUTOCONFIGURE runs automatically. If you have customized your booting sequence, make sure that AUTOCONFIGURE runs or that you explicitly configure all CIPCA devices before SYSTARTUP_VMS.COM exits.

- For CIPCA, CIXCD, and KFMSB, AUTOGEN will initially allocate 2 MB of bus addressable pool (BAP) for each CIPCA or CIXCD, and 4 MB for each KFMSB, as follows:

  - For systems with more than 1 GB of physical memory, OpenVMS creates a separate BAP. OpenVMS allocates this BAP memory for CIPCA, CIXCD, and KFMSB.

  - For systems with less than 1 GB of physical memory or with none of these devices, any bus addressable pool is merged with normal, nonpaged dynamic memory (nonpaged pool). In this case, the initial amount and maximum amount of nonpaged pool (as displayed by the DCL command SHOW MEMORY/POOL/FULL) does not match the value of the SYSGEN parameters NPAGEDYN and NPAGEVIR. Instead, the value of SYSGEN parameter NPAG_BAP_MIN is added to NPAGEDYN to determine the initial size, and the value of NPAG_BAP_MAX is added to NPAGEVIR to determine the maximum size.

    Your OpenVMS system may not require as much merged pool as the sum of these SYSGEN parameters. After your system has been running a few days, use AUTOGEN with feedback to fine-tune the amount of memory allocated for the merged, nonpaged pool and bus addressable pool.

These restrictions exist because SYS$PCADRIVER (the CIPCA driver) and SYS$PNDRIVER (the CIXCD and KFMSB driver) must be able to allocate physical memory guaranteed to have addresses below 1 GB.

4.15.2.4.5 MULTIPROCESSING and SYSTEM_CHECK Parameter in Systems with CIPCAs

The value of the SMP_SPINWAIT system parameter must be reset to 300000 (3 seconds) instead of the default 100000 (1 second) if you operate with either SYSTEM_CHECK set to 1 or MULTIPROCESSING set to either 1 or 2.

If you operate with either of these SYSTEM_CHECK or MULTIPROCESSING settings and do not change the value of SMP_SPINWAIT, a CIPCA adapter error could generate a CPUSPINWAIT system bugcheck similar to the following:

```
**** OpenVMS (TM) Alpha Operating System V6.2-1H3 - BUGCHECK ****
** Code=0000078C: CPUSPINWAIT, CPU spinwait timer expired
```

This restriction will be removed in a future OpenVMS release.

4.15.2.4.6 DECevent Required to Analyze Errors

Digital requires that you use DECevent to analyze error log files for the CIPCA adapter. The DCL command ANALYZE/ERROR_LOG has not been updated to support CIPCA and other new devices; using that command will result in improperly formatted error log entries.

Install the DECevent kit supplied on the OpenVMS Alpha Version 7.1 CD–ROM and then use the following commands to invoke DECevent to analyze dump files:

- DIAGNOSE — Analyzes the current system error log file
- DIAGNOSE *filename* — Analyzes the error log file named *filename.sys*

Use the HELP DIAGNOSE command to get more information about using DECevent.

4.15.2.5 MEMORY CHANNEL (Alpha Only)

V7.1

The following sections contain guidelines and restrictions that apply to MEMORY CHANNEL.

Note

If you are setting up a MEMORY CHANNEL cluster and need the hardware user's guide, you can copy it from the OpenVMS Version 7.1 CD–ROM using the following file name:

```
[DOCUMENTATION]HW_MEMORY_CHANNEL_UG.PS
```

You can set up the MEMORY CHANNEL software by making specific selections when running the CLUSTER_CONFIG.COM procedure. For more information about implementing MEMORY CHANNEL, see the MEMORY CHANNEL Technical Summary appendix in *Guidelines for OpenVMS Cluster Configurations*.

4.15.2.5.1 Memory Requirements

MEMORY CHANNEL consumes memory during normal operations. Each system in your MEMORY CHANNEL cluster must have at least 128 MB of memory.

4.15.2.5.2 Backup Interconnect Recommended for High Availability Configurations

MEMORY CHANNEL requires a central hub in configurations of three or more nodes. The MEMORY CHANNEL hub contains active, powered electronic components. In the event of a hub failure, either through a power shutdown or component failure, the MEMORY CHANNEL interconnect ceases operation. This type of failure does not occur with the other cluster interconnects, such as CI, DSSI, and most LAN configurations.

Digital therefore recommends that customers with MEMORY CHANNEL configurations who have high availability requirements consider using one of the following configurations to provide a second backup interconnect:

- In most cases a second interconnect can easily be configured by enabling the LAN (Ethernet or FDDI) for clustering. FDDI and 100 megabits-per-second Ethernet usually provide acceptable interconnect performance in the event of MEMORY CHANNEL failure. (See *OpenVMS Cluster Systems* and *Guidelines*

for OpenVMS Cluster Configurations for details on how to enable the LAN for clustering.)

- CI and DSSI interconnects automatically act as a backup for MEMORY CHANNEL.
- A configuration with two MEMORY CHANNEL interconnects provides the highest possible performance and continued operation if one MEMORY CHANNEL interconnect fails.

4.15.2.5.3 Adapter Support

OpenVMS Cluster Software Version 7.1 supports MEMORY CHANNEL using Version 1.0 MEMORY CHANNEL adapters (part number CCMAA–AA). These adapters have a bandwidth of approximately 35 megabytes per second. (The actual bandwidth depends on the system type and the read/write ratio.)

A higher performance MEMORY CHANNEL adapter (Version 1.5, part number CCMAA–BA), which provides approximately 55 megabytes per second bandwidth, can be ordered now and will be supported by OpenVMS Cluster Software Version 7.1 once you apply a remedial kit. This kit is expected to be available in March 1997. Contact your Digital support representative to obtain the kit.

4.15.2.5.4 MEMORY CHANNEL PCI Adapter Configuration Guidelines

This section contains system-specific guidelines and restrictions to use when configuring a system with a MEMORY CHANNEL PCI adapter.

- **AlphaServer 1000/1000A Requirements**

  To work with MEMORY CHANNEL, your console firmware must be at Rev 4.6 or higher. Digital recommends that you install the latest release of console firmware.

  A MEMORY CHANNEL PCI adapter can be installed in any of the three PCI slots of an AlphaServer 1000. For an AlphaServer 1000A, you can install a MEMORY CHANNEL PCI adapter in the highest three PCI slots (PCI 11, PCI 12, or PCI 13). The same is true for a second adapter.

- **AlphaServer 2000/2100 Requirements**

  - To work with MEMORY CHANNEL, your console firmware must be at Rev 4.3 or higher. Digital recommends that you install the latest release of console firmware.
  - Some older AlphaServer 2000/2100 systems may not have an I/O module that supports MEMORY CHANNEL. To check this, use the following procedure:
    + At the console prompt, enter the following:

```
P00>>> examine -b econfig:20008
econfig:            20008 04
```

    + If a hexadecimal value of 04 or greater is returned, as shown in the previous example, your I/O module supports MEMORY CHANNEL.

      If a value of less than 04 is returned, contact your Digital support representative for an upgrade. If you install the MEMORY CHANNEL hardware and power up on a system that is not hardware-ready for MEMORY CHANNEL, the following console error message is displayed:

```
*****************************************************
** Memory Channel hardware requirement ERROR  # 1     **
** See release notes, or Memory Channel User's Guide. **
*****************************************************
```

- A MEMORY CHANNEL PCI adapter can be installed in any of the three PCI slots in your AlphaServer 2000/2100 system (PCI 0, PCI 1, and PCI 2.)

- **AlphaServer 2100A Requirements**

  To work with MEMORY CHANNEL, your console firmware must be at Rev 4.3 or higher. Digital recommends that you install the latest release of console firmware.

  A MEMORY CHANNEL PCI adapter can be installed in any one of the bottom four PCI slots (PCI 4, PCI 5, PCI 6, and PCI 7) in your AlphaServer 2100A system. The same is true for a second adapter.

- **AlphaServer 4000/4100 Requirements**

  To work with MEMORY CHANNEL, your console firmware must be at Rev 2.0-3 or higher. This firmware version sets the PCI arbitration mode to round-robin, which is required by MEMORY CHANNEL. Please do not change this setting. Digital recommends that you install the latest release of console firmware.

  A MEMORY CHANNEL PCI adapter can be installed in any one of the PCI slots in your AlphaServer 4000/4100 system.

- **AlphaServer 8200/8400 Requirements**

  To work with MEMORY CHANNEL, your console firmware must be at Rev 2.3 or higher. Digital recommends that you install the latest release of console firmware.

  Install the MEMORY CHANNEL PCI adapter in PCI slots 0 through 7 of AlphaServer 8200/8400 systems.

  On AlphaServer 8200/8400 systems, there is a configuration restriction on the number of MEMORY CHANNEL PCI adapters that may be installed in a DLWPA card cage. If the DWLPA card cage contains an EISA bridge, an ISA bridge, or any other PCI devices, the maximum number of MEMORY CHANNEL PCI adapters that may be installed is one. If the DWLPA card cage contains only MEMORY CHANNEL PCI adapters, a maximum of two may be installed. This restriction does not apply to DWLPB card cages.

4.15.2.5.5 Rolling Upgrades

If MEMORY CHANNEL adapters (CCMAA-*xx*) have been added to the cluster *before* upgrading OpenVMS to Version 7.1, an MC_FORCEDCRASH bugcheck occurs on the first system when the second and subsequent systems perform AUTOGEN and SHUTDOWN during their Version 7.1 installations. This problem is caused by conflicting system parameters; it can be avoided by using one of the following procedures when upgrading:

- Upgrade all systems to OpenVMS Version 7.1 *before* adding the CCMAA-*xx* MEMORY CHANNEL adapters.

- If you have MEMORY CHANNEL hubs, power them off *before* upgrading each system to OpenVMS Version 7.1.

  After all systems have been upgraded to Version 7.1, power on the MEMORY CHANNEL hubs.

- If the nodes are connected directly in a virtual hub configuration, disconnect the BC12N cables *before* upgrading each system to OpenVMS Version 7.1.

  After all systems have been upgraded to Version 7.1, reconnect the BC12N cables.

4.15.2.5.6 MEMORY CHANNEL System Parameter Settings

This section contains a description for all MEMORY CHANNEL system parameters and recommendations for setting them. All numeric values in the following section are decimal numbers. Digital reserves the right to change the names of these parameters in a future release.

The system parameters specific to MEMORY CHANNEL are:

MC_SERVICES_P0 (Dynamic)
Controls whether other MEMORY CHANNEL nodes in the cluster continue to run if this node bugchecks or shuts down.

A value of 1 causes other nodes in the MEMORY CHANNEL cluster to crash with bugcheck code MC_FORCED_CRASH if this node bugchecks or shuts down.

The default value is 0. A setting of 1 is only intended for debugging purposes; the parameter should otherwise be left at its default value.

MC_SERVICES_P1 (Dynamic)
This parameter is reserved by Digital. Its value must be the same on all nodes connected by MEMORY CHANNEL.

MC_SERVICES_P2 (Static)
Specifies whether to load the PMDRIVER (PMA0) MEMORY CHANNEL cluster port driver.

PMDRIVER is a new driver that serves as the MEMORY CHANNEL cluster port driver. It works together with MCDRIVER (the MEMORY CHANNEL device driver and device interface) to provide MEMORY CHANNEL clustering. If PMDRIVER is not loaded, cluster connections will not be made over the MEMORY CHANNEL interconnect.

The default value for MC_SERVICES_P2 is 1. This default value causes PMDRIVER to be loaded when you boot the system. When you run CLUSTER_CONFIG.COM and select the MEMORY CHANNEL option, PMDRIVER is loaded automatically when you reboot the system.

Digital recommends that this value not be changed. This parameter value must be the same on all nodes connected by MEMORY CHANNEL.

MC_SERVICES_P3 (Dynamic)
Specifies the maximum number of tags supported. The maximum value is 2048 and the minimum value is 100.

The default value is 800. Digital recommends that this value not be changed.

This parameter value must be the same on all nodes connected by MEMORY CHANNEL.

MC_SERVICES_P4 (Static)
Specifies the maximum number of regions supported. The maximum value is 4096 and the minimum value is 100.

The default value is 200. Digital recommends that this value not be changed.

This parameter value must be the same on all nodes connected by MEMORY CHANNEL.

MC_SERVICES_P5 (Dynamic)
This parameter is reserved by Digital, and must remain at the default value of 8000000. This value must be the same on all nodes connected by MEMORY CHANNEL.

MC_SERVICES_P6 (Static)
Specifies MEMORY CHANNEL message size, the body of an entry in a free queue, or a work queue. The maximum value is 65536 and the minimum value is 544. The default value is 992. This value is suitable in all cases, except for systems with highly constrained memory.

For such systems, you can reduce the memory consumption of MEMORY CHANNEL by slightly reducing the default value of 992. This value must always be equal to or greater than the result of the following calculation:

- Select the larger of SCS_MAXMSG and SCS_MAXDG
- Round that value up to the next quadword

This parameter value must be the same on all nodes connected by MEMORY CHANNEL.

MC_SERVICES_P7 (Dynamic)
Specifies whether to suppress or display messages about MEMORY CHANNEL activities on this node. Can be set to a value of 0, 1, or 2.

A value of 0 indicates nonverbose mode - no informational messages will appear on the console or in the error log.

A value of 1 indicates verbose mode—informational messages from both MCDRIVER and PMDRIVER will appear on the console and in the error log.

A value of 2 provides the same output as a value of 1, with the addition of PMDRIVER stalling and recovery messages.

The default value is 0. Digital recommends that this value not be changed except while debugging MEMORY CHANNEL problems or adjusting the MC_SERVICES_P9 parameter.

MC_SERVICES_P8 (Static)
This parameter is reserved by Digital, and must remain at the default value of 0. The value must be the same on all nodes connected by MEMORY CHANNEL.

MC_SERVICES_P9 (Static)
Specifies the number of initial entries in a single channel's free queue. The maximum value is 2048 and the minimum value is 10.

Note that MC_SERVICES_P9 is not a dynamic parameter; you must reboot the system after each change in order for that change to take effect.

Default value is 150. Digital recommends that this value not be changed.

This parameter value must be the same on all nodes connected by MEMORY CHANNEL.

Setting the SYSTEM_CHECK and MULTIPROCESSING System Parameters

To avoid a CPUSPINWAIT problem, Digital recommends the following settings for system parameters:

Parameter	Setting
SYSTEM CHECK	0 (default is 0)
MULTIPROCESSING	3 or 4 (default is 3)

With these two parameters set at the recommended values, the SYSTEM_SYNCHRONIZATION_MIN version of the SYSTEM_SYNCHRONIZATION execlet is loaded. This avoids a problem that arises when the SYSTEM_SYNCHRONIZATION_MON form of the execlet is loaded.

If you have reason to set SYSTEM_CHECK and MULTIPROCESSING to different values from those recommended, you can reduce the likelihood of encountering a problem by setting the SMP_LNGSPINWAIT system parameter to a value of 9000000.

NPAGEVIR Parameter on Systems with Very Large Memory

On systems containing very large amounts of nonpaged pool memory, the MEMORY CHANNEL may be unable to complete initialization. The symptom is a repetitive message "Hub timeout - reinitializing adapter" on the console. To fix this problem, examine the value of the SYSGEN parameter NPAGEVIR. If its value is greater than 1 gigabyte, consider lowering it to about half of that. Thereafter, a reboot of your system should allow the MEMORY CHANNEL to complete initialization.

4.15.2.6 System Startup in an OpenVMS Cluster Environment (Alpha Only)

V6.2

In an OpenVMS Cluster environment on Alpha systems, under some circumstances the system startup procedure may fail to write a new copy of the ALPHAVMSSYS.PAR file. If this occurs, the console output from the boot sequence reports the following messages:

```
%SYSGEN-E-CREPARFIL, unable to create parameter file
-RMS-E-FLK, file currently locked by another user
```

This error creates an operational problem only when changing SYSGEN parameters using a conversational boot. For a normal, nonconversational boot, this error message is purely cosmetic because the parameter file has not changed. If a conversational boot is used, and SYSGEN parameters are changed at boot time, these changed parameters will be correctly used for the current boot of the system. However, since the boot procedure does not successfully write a new copy of the parameter file, these changed parameters will not be used in subsequent boots.

To permanently change SYSGEN parameters that have been changed by a conversational boot, run SYSGEN after the system has completed booting, and execute the following commands:

```
SYSGEN> USE ACTIVE
SYSGEN> WRITE CURRENT
```

4.15.3 Documentation Changes and Corrections

The following notes describe corrections to cluster documentation.

4.15.3.1 *Guidelines for OpenVMS Cluster Configurations*

V7.1

In Figure A–9 in *Guidelines for OpenVMS Cluster Configurations*, the labels for the differential host adapters should be KZPSA-BB, not KZPSA-AA. The KZPSA-AA is not available.

4.15.3.2 *OpenVMS System Manager's Manual*

V7.1

In Figure 20–2 of the *OpenVMS System Manager's Manual*, the CLUSTER Report section was inadvertently omitted from the hardcopy and Bookreader versions of the manual. Figure 4–1 contains the information that was missing in the manual.

Figure 4–1 SHOW CLUSTER Display with CLUSTER Report

View of Cluster from system ID 65536 node: CLUB 31–DEC–1996 14:00:00

SYSTEMS		MEMBERS
NODE	SOFTWARE	STATUS
CLUB	VMS V6.2	MEMBER
HSJ400	HSJ V25J	
HSC900	HSC V860	
CHIP	VMS V7.1	MEMBER
DISK3	RFX V256	
DISK1	RFX V256	
SPREE	VMS V6.2	MEMBER
SPRITZ	VMS V7.1	MEMBER

CLUSTER						
CL_EXP	CL_QUORUM	CL_VOTES	QF_VOTE	CL_MEMBERS	FORMED	LAST_TRANSITION
3	2	3	NO	4	15–JUN–1996	10–DEC–1996

ZK–8999A–GE

4.16 POLYCENTER Software Installation Utility

The release notes in this section pertain to the POLYCENTER Software Installation utility. Also see Section 5.16 for notes about this utility that are of interest to programmers.

4.16.1 Changes and Enhancements

This section describes changes to the POLYCENTER Software Installation utility and the DCL interface for the PRODUCT command.

4.16.1.1 PRODUCT Command Changes

V7.1

Changes to the PRODUCT command are described in the following sections.

4.16.1.1.1 New /INTERFACE Qualifier Invokes Motif Interface

All functions of the POLYCENTER Software Installation utility are available from the DCL interface through a set of PRODUCT commands. In addition, workstation users can perform a subset of these functions through an optional Motif interface. Starting with Version 7.1, the method of accessing the Motif interface has been changed to use a new /INTERFACE qualifier. (The Motif interface will be retired with the next OpenVMS release.)

The default is /INTERFACE=CHARACTER_CELL. You now must explicitly specify /INTERFACE=DECWINDOWS in order to invoke the Motif interface on a workstation:

```
$ PRODUCT /INTERFACE=DECWINDOWS
```

If you do not specify /INTERFACE=DECWINDOWS, the DCL interface is used by default. As of Version 7.1, full prompting for PRODUCT command parameters is now provided when you enter only the PRODUCT command at the DCL prompt.

4.16.1.1.2 Qualifiers Removed from Documentation

Prior to OpenVMS Version 7.1, the following qualifiers were mistakenly included in the *OpenVMS System Management Utilities Reference Manual* and in DCL help:

- The /[NO]KEEP qualifier for the PRODUCT INSTALL command.
- The /[NO]TEST qualifier for the PRODUCT RECONFIGURE command. (Note that this qualifier is supported for the PRODUCT INSTALL command.)

The documentation has been corrected to remove these qualifiers.

4.16.1.1.3 Qualifiers Retired

Prior to OpenVMS Version 7.1, the /DEVICE and /DIRECTORY qualifiers for the PRODUCT SHOW OBJECT command were documented in the *OpenVMS System Management Utilities Reference Manual* and in DCL help, but they did not function as described. These qualifiers are no longer supported.

The documentation has been corrected to remove these qualifiers. Digital expects to enhance the PRODUCT SHOW OBJECT command in a future release and may provide the features of these qualifiers in a different manner.

4.16.2 Problems and Restrictions

Notes in this section pertain to problems and restrictions with using the POLYCENTER Software Installation utility to install, remove, and reconfigure software products. Problems and restrictions of interest to programmers are described in Section 5.16.1.

4.16.2.1 PRODUCT Command Lacks Options to Control Output

V6.2

Commands such as PRODUCT FIND and PRODUCT SHOW that can display more than a screen of text do not provide qualifiers such as /PAGE to control scrolling or /OUTPUT to redirect output to a file. As a result, information can scroll off the screen. Digital expects to enhance these commands in a future release.

4.16.2.2 Product Removal Restrictions

V6.1

Removing a product using the POLYCENTER Software Installation utility results in the removal of accounts created for that product. This happens regardless of whether the SYSUAF.DAT file is shared by another system disk.

The same problem exists with rights identifiers and the file RIGHTSLIST.DAT.

4.16.3 Corrections

The following note describes a correction to the POLYCENTER Software Installation utility.

4.16.3.1 Estimated Disk Space Requirements

V7.1

At the beginning of the execution phase of a PRODUCT INSTALL, PRODUCT REMOVE, or PRODUCT RECONFIGURE command, the POLYCENTER Software Installation utility estimates the amount of disk space required to perform the operation. It estimates both the peak space utilization (maximum number of disk blocks used at once) and the net space utilization (the number of disk blocks consumed at the end of the operation).

Previously, these estimates were often too low, causing an installation to fail even though the utility reported that enough disk space was available. These problems have been corrected in OpenVMS Version 7.1 by implementing a more comprehensive algorithm.

Informational messages that convey the disk space estimates are no longer displayed by default. Starting with OpenVMS Version 7.1, you must specify the /LOG qualifier to obtain this information. In addition, the format of the data has changed. For example, execution of the PRODUCT INSTALL /LOG command for product TEST displays the following:

```
Execution phase starting ...
The following product will be installed:
    DEC AXPVMS TEST V1.0
%PCSI-I-VOLINFO, estimated disk block usage for volume DISK$ALPHAVMS
-PCSI-I-VOLSPCFB,     free space before operation:      871134
-PCSI-I-VOLSPCPK,     peak space utilization:             3705
-PCSI-I-VOLSPCNT,     net space utilization:              2982
-PCSI-I-VOLSPCFA,     free space after operation:       868152
```

4.17 RMS Journaling

The following release notes pertain to RMS Journaling for OpenVMS.

4.17.1 Problems and Restrictions

The following notes describe restrictions to RMS Journaling for OpenVMS.

4.17.1.1 Remote Access of Recovery Unit Journaled Files in an OSI Environment

V6.1

OSI nodes that host recovery unit journaled files that are to be accessed remotely from other nodes in the network must define SYS$NODE to be a Phase IV style node name. The node name specified by SYS$NODE must be known to any remote node attempting to access the recovery unit journaled files on the host node, and must be sufficiently unique for the remote node to use this node name to establish a DECnet connection to the host node. This restriction applies only

to recovery-unit journaled files accessed across the network in an OSI or mixed OSI and non-OSI environment.

4.17.1.2 VFC Format Sequential Files

VAX V5.0
Alpha V1.0

You cannot update variable fixed-length control (VFC) sequential files when using before-image or recovery unit journaling. The VFC sequential file format is indicated by the symbolic value FAB$C_VFC in the FAB$B_RFM field of the FAB.

4.18 Security

This section contains release notes pertaining to system security.

4.18.1 Changes and Enhancements

This section describes changes and enhancements to system security.

4.18.1.1 DETACH Privilege Renamed to IMPERSONATE

V7.1

The DETACH privilege has been renamed IMPERSONATE. The power of this privilege has increased over time so that a user with this privilege can now perform a general impersonation of another user.

Some ways that the IMPERSONATE privilege can be used are as follows:

- You can use the DCL command RUN/DETACH/UIC to create a detached process running under another user's identity. (See the *OpenVMS DCL Dictionary* for details.)
- You can use the DCL command SUBMIT/USER to create a batch job running under another user's identity. (See the *OpenVMS DCL Dictionary* for details.)
- You can use the $PERSONA_CREATE system service to create a security profile that an application can use to impersonate another user's identity. (See the *OpenVMS System Services Reference Manual* for details.)

4.18.1.2 DIRECTORY Command Now Summarizes Suppressed PATHWORKS ACEs

V7.1

In OpenVMS Version 7.1, if you execute the DCL command DIRECTORY /SECURITY or DIRECTORY/FULL for files that contain PATHWORKS access control entries (ACEs), the hexadecimal representation for each PATHWORKS ACE is no longer displayed. Instead, the total number of PATHWORKS ACEs encountered for each file is summarized in this message: "Suppressed *n* PATHWORKS ACE." To display the suppressed PATHWORKS ACEs, use the DUMP/HEADER command.

4.19 System Management Utility

This section contains release notes pertaining to the OpenVMS System Management utility (SYSMAN).

4.19.1 Corrections

The following sections describe corrections to problems associated with SYSMAN.

All of these problems have been fixed in OpenVMS Version 7.1. Some corrections have been incorporated into remedial kits for previous versions of OpenVMS. Contact your Digital support representative or your authorized reseller for more information.

4.19.1.1 DO Command Performance Corrected

V7.1

Previously, the SYSMAN DO command performed slowly for certain DCL commands that generate no output (for example, the DEFINE command). This problem has been fixed.

4.19.1.2 DO Command Output Line Length Is Again 2048 Bytes

V7.1

The maximum output line length with the SYSMAN DO command was previously changed from 2048 to 512 bytes. The change back to 2048 bytes allows the correct display of files created by PATHWORKS when you use DCL commands such as DIRECTORY/SECURITY.

4.19.1.3 SHUTDOWN.LOG File Now Closes Properly With SHUTDOWN NODE Command

V7.1

Previously, when the SYSMAN SHUTDOWN NODE was executed, the output file for the shutdown (SYS$MANAGER:SHUTDOWN.LOG) was not properly closed. This problem has been fixed.

4.19.1.4 Extraneous Message Removed on PARAMETER SET Command

V7.1

Previously, the SYSMAN PARAMETER SET command printed out an extraneous SYSTEM-W-NOMORENODE message. This problem has been fixed.

4.19.1.5 Exclamation Mark at Start of Command Line No Longer Causes Warning Message

V7.1

Previously, if a SYSMAN command line started with an exclamation mark (!) character (to denote the start of a comment), a CLI-W-NOCOMD message was output. This problem has been fixed.

4.19.1.6 SYSMAN DISKQUOTA ENABLE and DISABLE Commands Now Work On Entire Cluster (VAX Only)

V7.1

On VAX systems, the SYSMAN DISKQUOTA ENABLE and DISABLE commands did not work over the entire cluster if the disk was mounted clusterwide. For example, if you issued an ENABLE command, some of the nodes were enabled and others were not. This problem has been fixed so that these commands now work over the entire cluster.

This problem did not affect SYSMAN on Alpha systems.

4.20 System Parameters

This section contains release notes pertaining to OpenVMS system parameters.

4.20.1 Changes and Enhancements

The following sections describe changes to or clarifications about several system parameters. Consult online help for SYSGEN to obtain complete details about system parameters.

4.20.1.1 DUMPSTYLE

V7.1

DUMPSTYLE specifies the method of writing system dumps.

DUMPSTYLE is a 32-bit mask, with the following bits defined. Each bit can be set independently. The value of the system parameter is the sum of the values of the bits that have been set. Remaining or undefined values are reserved to Digital.

Bit	Mask	Description	
0	00000001	0 =	Full dump (SYSGEN default). The entire contents of physical memory will be written to the dump file.
		1 =	Selective dump. The contents of memory will be written to the dump file selectively to maximize the usefulness of the dump file while conserving disk space.
1	00000002	0 =	Minimal console output.
		1 =	Full console output (includes stack dump, register contents, and so on).
2	00000004	0 =	Dump to system disk.
		1 =	Dump off system disk (DOSD) to an alternate disk.[1] (Refer to the *OpenVMS System Manager's Manual* for details.)
3 (Alpha only)[2]	00000008	0 =	Do not compress.
		1 =	Compress. (See description following this table.)
4 — 14			Reserved to Digital.
15 (VAX only)[3]	00008000	0 =	Disable use of bits 16 — 27.
		1 =	Enable use of bits 16 — 27.
16 — 27 (VAX only)[3]	0FFF0000		Range of DOSD unit numbers.
28 — 31			Reserved to Digital.

[1]If you plan to enable the Volume Shadowing Version 7.1 new feature that enables minimerge on an Alpha system disk, be sure to specify DOSD to an alternate disk. See Section 4.24.2.4 for more information.

[2]VAX systems do not support dump compression.

[3]Specific to VAX 7000 systems.

On Alpha systems, system managers can save space on the system disk and, in the event of a crash, save time recording the system memory by using the OpenVMS Alpha dump compression feature. Unless the system manager

overrides the default AUTOGEN calculations (by setting DUMPSTYLE in MODPARAMS.DAT), AUTOGEN uses the following algorithm:

- On a system with less than 128 MB of memory, the system sets the DUMPSTYLE to 1 (a raw selective dump) and sizes the dump file appropriately.
- On a system with 128 MB of memory or greater, the system sets the DUMPSTYLE to 9 (a compressed selective dump), and creates the dump file at two-thirds the value of the corresponding raw dump.

Examples:

The mask of 00000006 directs the system to send a full dump, with full console output, off the system disk (to the alternate disk).

For a VAX 7000, a mask of 00098006 directs the system to send a full dump with full console output to the DOSD whose unit number is 9.

On Alpha systems, the mask of 00000009 directs the system to compress a selective dump with minimal console output.

DUMPSTYLE has AUTOGEN and DYNAMIC attributes.

4.20.1.2 GBLPAGES and GBLPAGFIL (Alpha only)

V7.1

As of OpenVMS Alpha Version 7.1, the system parameters GBLPAGES and GBLPAGFIL have been modified to become dynamic parameters. Users with the CMKRNL privilege can now change these parameter values on a running system. Increasing the value of the GBLPAGES parameter will allow the global page table to expand, on demand, up to the new maximum size.

For more information about using the GBLPAGES and GBLPAGFIL parameters for VLM applications, see *OpenVMS Alpha Guide to 64-Bit Addressing and VLM Features*.

4.20.1.3 GH_RSRVPGCNT (Alpha Only)

V7.1

On Alpha systems, GH_RSRVPGCNT specifies the number of pages in the resident image granularity hint region that the Install utility can use after the system has finished booting.

If bit 2 of the LOAD_SYS_IMAGES parameter is set, the image LDR$WRAPUP releases all unused pages in the granularity hint region at the the end of system startup. The unused pages of the resident image granularity hint region are either reserved for future use or given back to the free memory list.

GH_RSRVPGCNT specifies the number of pages that LDR$WRAPUP attempts to leave in the resident image granularity hint region. If the GH_RSRVPGCNT number of pages is larger than the unused pages in the granularity hint region, the region is not expanded to accommodate the number of pages requested.

GH_RSRVPGCNT has the FEEDBACK attributes.

4.20.1.4 LOCKIDTBL_MAX

V7.1

This parameter is obsolete as of OpenVMS Version 7.1. Beginning with OpenVMS Version 7.1, the lock id table continually expands if free memory is available.

4.20.1.5 MAXBUF (VAX Only)

V7.1

On OpenVMS VAX Version 7.1 systems, the default value of the MAXBUF system parameter has been increased to 4112 from 2064. The default value on Alpha systems remains unchanged at 8192.

4.20.1.6 PIOPAGES

V7.1

PIOPAGES specifies the size of the process I/O segment, which holds data structures and buffer pool space for RMS to use when handling I/O involving process-permanent files.

PIOPAGES is now a dynamic parameter. Once PIOPAGES has been reset in SYSGEN, any new process will receive the changed value.

PIOPAGES has the AUTOGEN and DYNAMIC attributes.

4.20.1.7 RMS_DFMBC

V7.1

RMS_DFMBC specifies a default multiblock count only for record I/O operations, where *count* is the number of blocks to be allocated for each I/O buffer.

You can set this system parameter with the DCL command SET RMS_DEFAULT /SYSTEM and display the parameter using the SHOW RMS_DEFAULT command.

RMS_DFMBC has the AUTOGEN and DYNAMIC attributes.

4.20.1.8 RMS_GBLBUFQUO

V7.1

This parameter is obsolete as of OpenVMS Version 7.1. The number of global buffers on a system can be sufficiently controlled by the system parameters GBLPAGES, GBLPAGFIL, and GBLSECTIONS.

4.20.1.9 STARTUP_P1–8

V7.1

STARTUP_P1 specifies the type of system boot the system-independent startup procedure is to perform. " " indicates a full boot. "MIN" indicates a minimum boot that starts only what is absolutely necessary for the operating system to run.

STARTUP_P2 controls whether verification is set during the execution of the system-independent startup procedure. " " indicates no verification. "TRUE" indicates that verification is enabled.

STARTUP_P3 through STARTUP_P8 are reserved for future use.

4.20.1.10 VIRTUALPAGECNT (VAX Only)

V7.0

On VAX systems, VIRTUALPAGECNT sets the maximum number of virtual pages that can be mapped for any one process. A program is allowed to divide its virtual space between the P0 and P1 tables in any proportion.

If you use SYS$UPDATE:LIBDECOMP.COM to decompress libraries and the VIRTUALPAGECNT setting is low, make sure you set the PGFLQUOTA field in the user authorization file to at least twice the size of the library.

At installation time, AUTOGEN automatically sets an appropriate value for VIRTUALPAGECNT. The value depends on the particular configuration — the type and number of graphics adapters on the system, if any exist. You cannot set the VIRTUALPAGECNT value below the minimum value required for your graphics configuration.

Because the VIRTUALPAGECNT setting is used to support hardware address space rather than system memory, do not use the value of VIRTUALPAGECNT that AUTOGEN sets to gauge the size of your page file.

VIRTUALPAGECNT has the AUTOGEN, GEN, and MAJOR attributes.

4.20.1.11 VIRTUALPAGECNT (Alpha Only)

V7.0

Starting with OpenVMS Alpha Version 7.0, VIRTUALPAGECNT ceases to be a tunable parameter on Alpha systems and is no longer used to specify the virtual size of a process. The process page tables have migrated from system space to a dedicated page table space that is guaranteed to be large enough to accommodate the virtual address space provided by the system. This migration has rendered the parameter obsolete, and OpenVMS Alpha ignores its contents entirely.

Note, however, that on Alpha systems the parameter remains in existence for compatibility purposes and now has a default and maximum value of %X7FFFFFFF. Both SYSBOOT and AUTOGEN enforce this default value.

VIRTUALPAGECNT has the AUTOGEN, GEN, and MAJOR attributes.

4.20.2 Documentation Changes and Corrections

This section describes a correction to the documentation of the MSCP_CMD_TMO system parameter.

4.20.2.1 *OpenVMS Version 7.1 New Features Manual*

V7.1

In Section 3.16.6 of the *OpenVMS Version 7.1 New Features Manual*, the default value for the MSCP_CMD_TMO system parameter is incorrectly documented as 600. The correct default value is 0. A nonzero setting increases the amount of time before an MSCP command times out.

The rest of the description in the *OpenVMS Version 7.1 New Features Manual* is correct.

4.21 Terminal Fallback Facility (TFF) (Alpha Only)

On OpenVMS Alpha systems, the Terminal Fallback facility (TFF) includes a fallback driver (SYS$FBDRIVER.EXE), a shareable image (TFFSHR.EXE), a terminal fallback utility (TFU.EXE), and a fallback table library (TFF$MASTER.DAT).

Note

TFFSHR has been removed from IMAGELIB because it is not a documented, user-callable interface. The image is still available in the SYS$LIBRARY: directory.

To start TFF, invoke the TFF startup command procedure located in SYS$MANAGER, as follows:

```
$ @SYS$MANAGER:TFF$SYSTARTUP.COM
```

To enable fallback or to change fallback characteristics, invoke the Terminal Fallback utility (TFU), as follows:

```
$ RUN SYS$SYSTEM:TFU
TFU>
```

To enable default fallback to the terminal, issue the following DCL command:

```
$ SET TERMINAL/FALLBACK
```

OpenVMS Alpha TFF differs from OpenVMS VAX TFF in the following ways:

- On Alpha systems, the TFF fallback driver is named SYS$FBDRIVER.EXE. On VAX systems, the TFF fallback driver is named FBDRIVER.EXE.
- On Alpha systems, TFF is capable of handling 16-bit character fallback. The OpenVMS Alpha fallback table library (TFF$MASTER.DAT) contains four more 16-bit character tables than the VAX library. Table 4–1 describes these additional tables.

Table 4–1 TFF Character Fallback Tables

Table Name	Base	Description
BIG5_HANYU	BIG5	BIG5 for CNS 11643 (SICGCC) terminal/printer
HANYU_BIG5	CNS	CNS 11643 (SICGCC) for BIG5 terminal/printer
HANYU_TELEX	CNS	CNS 11643 for MITAC TELEX-CODE terminal
HANGUL_DS	KS	KS for DOOSAN 200 terminal

  These tables are used mainly by the Asian region. Also, the table format was changed due to the support of 16-bit character fallback.

- On Alpha systems, the TFU command SHOW STATISTICS does not display the size of the fallback driver (SYS$FBDRIVER.EXE).

RT terminals are not supported by TFF.

For more information about the Terminal Fallback facility, refer to the *OpenVMS Terminal Fallback Utility Manual*[1].

4.22 UETP (User Environment Test Package)

This section contains notes pertaining to UETP.

4.22.1 Problems and Restrictions

The following note describes a support restriction for this release.

[1] This manual has been archived but is available in PostScript and DECW$BOOK (Bookreader) formats on the OpenVMS Documentation CD–ROM. A printed book can be ordered through DECdirect (800–354–4825).

4.22.1.1 RRD45 CD–ROM Testing (VAX Only)

V7.0

UETP does not support the RRD45 CD–ROM on OpenVMS VAX Version 7.0 and later. This release note describes a temporary workaround that you can use until support is added in a future release.

If you have an RRD45 CD–ROM, UETP will fail in the following manner:

```
%UETP-S-BEGIN, UETINIT01 beginning at 13-NOV-1995 16:35:22.76
 .
 .
    **********************
    *  DISK_NODE$DKA     *
    *  Error count =  1  *
    **********************
    -UETP-E-TEXT, RMS file error in file
    NODE$DKA500:NODE_NODE$DKA5000.TST
    -RMS-E-DNR, device not ready, not mounted, or unavailable
    %UETP-S-ENDED, UETDISK00 ended at 13-NOV-1995 16:36:15.10
 .
 .
    NODE$DKA:   testable     100, 200, 300
                untestable   500
 .
 .
 ************************************************
 *                                              *
    END OF UETP PASS 1 AT 13-NOV-1995 16:44:49.85
 *                                              *
 ************************************************
```

You can work around this problem and test the RRD45 device by running the CD–ROM test (UETCDRO00.EXE) against the RRD45 separately from UETP.COM. Follow these steps:

1. Edit UETSUPDEV.DAT to include the following line:

   ```
   01 36 UETCDRO00.EXE ! RRD45
   ```

2. Edit UETINIDEV.DAT to change the "N" in the row for the RRD45 device to a capital "T". The line should look like this (the number might be different):

   ```
   UCB T 00500 UETCDRO00.EXE
   ```

3. Run the CD–ROM test, for example:

   ```
   $ RUN UETCDRO00.EXE
   Controller designation?: NODE$DKA
   %UETP-S-BEGIN, UETCDRO00 beginning at 13-NOV-1995 15:15:43.20
   %UETP-I-ABORTC, CDRO_NODE$DKA to abort this test, type ^C
   %UETP-S-ENDED, UETCDRO00 ended at 13-NOV-1995 15:18:45.98
   $
   ```

4.23 VAX System Dump Analyzer (SDA)

The following release notes pertain to the VAX System Dump Analyzer (SDA).

4.23.1 Changes and Enhancements

The following note describes a change to VAX SDA.

4.23.1.1 Dump File Process

V7.0

OpenVMS VAX SDA now uses RMS file access to process the dump file instead of mapping the dump file into the working set. Because of this change, some commands that execute over a large range, such as SEARCH, may take longer.

Another effect of this change is that a value of 16,000 for the system parameter VIRTUALPAGECNT should be sufficient to analyze any dump, even if a large number of symbols is read in.

4.24 Volume Shadowing

The following release notes pertain to volume shadowing software.

4.24.1 Changes and Enhancements

This section describes changes to volume shadowing software.

4.24.1.1 Shadow Set Member Support Raised to 500

V7.1

As of OpenVMS Version 7.1, the limit on the number of supported shadow set members has been raised from 400 to 500.

4.24.1.2 Per-Disk Licensing Enforced

V7.1

Volume Shadowing Version 7.1 includes a license check for each disk that is shadowed using the per-disk volume shadowing license. Described as follows, this feature provides system managers with an easy and effective method for controlling these licenses.

Per-disk volume shadowing licenses apply to full shadow set members only. When the number of shadow set members exceeds the number of per-disk licenses for five minutes, shadowing issues an OPCOM warning message. You can have this message also sent to an E-mail account by defining the system logical SHADOW_SERVER$MAIL_NOTIFICATION to a standard OpenVMS Mail address or a UNIX (Internet) address. An invalid address will not generate a failure message.

Shadowing issues notification again 59 minutes after non-compliant shadow set members are mounted. One minute later, shadow set members are automatically removed from shadow sets until the number of members equals the number of licenses. Members are removed systematically from multiple-member shadow sets; single-member shadow sets are not affected.

Disks that are the target of a copy operation do not consume a license unit until the copy is complete. Thus, it is always possible to obtain a copy of a single-member shadow set.

4.24.1.3 Volume Shadowing Locally Connected SCSI Disks (VAX Only)

V6.2

The VAX SCSI disk driver (DKDRIVER) does not implement the same level of error handling that exists in all other OpenVMS disk drivers. Consequently, the OpenVMS Volume Shadowing product is unable to recover from several rare error conditions when used on SCSI disks that are locally connected to VAX 3000 and VAX 4000 series systems. This problem does not occur on CI-based or DSSI-based SCSI storage.

Problems resulting from the unimplemented error handling typically are shadow set hangs or system crashes. No evidence of data corruption has been seen in Digital test labs. No hangs or crashes at customer sites have been reported. The complexity of this problem, which has existed since Version 5.5-2, precludes it being fixed with a patch, and Digital has no plans to implement the missing error handling code.

Customers who wish to use OpenVMS Volume Shadowing on SCSI disks that are locally connected to VAX 3000 and 4000 series systems will be supported by Digital up to the limits of the current error handling capabilities. Volume Shadowing will continue to deliver data availability in the event of disk media failure on these systems, although some errors will cause a system failure, necessitating a reboot.

4.24.2 Problems and Restrictions

The following sections describe known problems and other considerations pertaining to volume shadowing.

4.24.2.1 Incompatibility With StorageWorks RAID Software

V7.1

An incompatibility exists between StorageWorks RAID Software and the enhanced volume shadowing provided in both OpenVMS Version 7.1 and in the Cluster Compatibility Kit. Because of this incompatibility, RAID software can no longer detect a shadow set state change.

If your system uses StorageWorks RAID 0+1 RAID sets and volume shadowing, contact your Digital support representative for a new driver before installing either OpenVMS Version 7.1 or the Cluster Compatibility Kit for Version 6.2 systems.

Note

This incompatibility does not affect RAID 0 (without shadowing) arrays or RAID 5 arrays.

4.24.2.2 Bad Block Repair (BBR) Logic Problem

V7.1

The OpenVMS Volume Shadowing driver's Bad Block Repair (BBR) logic does not perform correctly in some cases.

A partial solution for this BBR problem has been available in remedial kits for Version 6.2 and prior versions.

The complete solution, which includes enhanced code for the COPY DATA repair operations, was developed too late to be included in Version 7.1, but it is available in a remedial kit for Version 7.1.

If you update any Version 6.2 systems with the Cluster Compatibility Kit (see Section 4.15.1.2), the new Volume Shadowing drivers in that kit supersede any drivers with the partial BBR solution that were shipped in earlier remedial kits for Version 6.2. The Cluster Compatibility Kit does not include the BBR solution, but you can obtain the complete solution in a new remedial kit.

Contact your Digital support representative to obtain any of these remedial kits.

4.24.2.3 HSD10 Virtual Disks

V7.1

The HSD10 controller supports a virtual disk capability by partitioning physical disks. To prevent data corruption, do not use OpenVMS Volume Shadowing to create shadow sets using HSD10 virtual disks.

4.24.2.4 Minimerge Capability Requires Dump Off System Disk

V7.1

With the use of the new DUMPSTYLE system parameter in OpenVMS Alpha Version 7.1, you can now enable minimerge on shadowed system disks. The DUMPSTYLE parameter lets you specify that the system write a dump to a nonshadowed, nonsystem disk of your choice. This results in a considerable performance improvement.

Control the shadowing of a system disk by setting the SHADOW_SYS_DISK system parameter as shown in Table 4–2.

Table 4–2 SHADOW_SYS_DISK System Parameter Settings

Setting	Description
0	Do not shadow the system disk
1	Shadow the system disk; disable system disk minimerge
4097	Shadow the system disk; enable system disk minimerge

Do not enable system disk minimerge without using the DUMPSTYLE parameter to dump off system disk, as described in Section 4.20.1.1. If you do not set the DUMPSTYLE parameter to DOSD and proceed to enable minimerge for system disks, minimerge will be activated, to the detriment of crash dump analysis.

In the event that you accidentally enable minimerge to a system disk that receives a crash dump and you have not set up dump off system disk, you can recover if you know which disk contains the valid dump. Remove that member, remount it, and remove the dump from that member.

4.24.2.5 Minimerge Version Incompatibility

V7.1

The Volume Shadowing software for OpenVMS Version 7.1 has been revised with substantial quality improvements. However, this creates an incompatibility in the minimerge function between Version 7.1 and Version 6.2 nodes within the same cluster. In such configurations, the Volume Shadowing software detects this incompatibility during Version 7.1 installation and disables the minimerge capability for the entire cluster.

This problem is resolved by installing the OpenVMS Cluster Compatibility Kit that ships with OpenVMS Version 7.1 (see Section 4.15.1.2). This kit makes the Version 6.2 minimerge function compatible with that of nodes running Version 7.1 software.

4.24.2.6 HSZ40 and Transportable SCSI Disk Shadow-Set Members

V6.2

An HSZ40 Raid-Array Controller provides the capability of an OpenVMS initialized SCSI disk (that is, one with a Files–11 ODS-2 format on it) to be moved between an OpenVMS controlled SCSI bus and an HSZ40 controlled SCSI bus without reinitializing the disk and losing data. Disks that contain this functionality are called **transportable** disks.

A SCSI disk initialized by the HSZ40 and then subsequently initialized by OpenVMS is called a **nontransportable** disk, and cannot be moved to an OpenVMS controlled SCSI bus without losing data.

OpenVMS Volume Shadowing requires that a SCSI disk support the SCSI commands READ_LONG/WRITE_LONG. These SCSI commands in conjunction with OpenVMS Volume Shadowing are used to handle certain classes of errors as seen under normal volume shadowing operations. SCSI disks that support the READ_LONG/WRITE_LONG capability while connected to an OpenVMS controlled SCSI bus, lose this capability when the transportable disks are moved to the SCSI bus controlled by an HSZ40.

The lack of READ_LONG/WRITE_LONG capability is detected at shadow-set MOUNT time, by the following error:

```
MOUN$_DEVNOFE, device does not support FORCED ERROR handling
```

To correct this problem, specify the MOUNT qualifier /OVERRIDE=NO_FORCED_ERROR at shadow-set MOUNT time.

Note that specifying this MOUNT qualifier may cause shadow-set member SCSI disks to be removed from a shadow set if certain error conditions arise that cannot be corrected.

Digital recommends that HSZ40 nontransportable SCSI disks be used to contain shadow-set members that support READ_LONG/WRITE_LONG functionality, and offer benefits provided by the level of RAID chosen at initialization time.

4.24.3 Corrections

The following notes describe corrections to former volume shadowing problems.

4.24.3.1 Crash Dump Problem Corrected (Alpha Only)

V7.1

Formerly, if a boot device was removed from a multiple-member system disk shadow set on an OpenVMS Alpha system, this removal resulted in the loss of a crash dump if a subsequent system failure occurred.

This problem has been corrected in OpenVMS Alpha Version 7.1.

4.24.3.2 DISMOUNT/CLUSTER Command Problem Corrected

V7.1

Formerly, issuing the DISMOUNT/CLUSTER command to a shadow set sometimes caused a clusterwide hang. This problem has been corrected in OpenVMS Version 7.1.

5
Programming Release Notes

This chapter provides release notes about both application and system programming on OpenVMS systems.

For information about new programming features included in OpenVMS Version 7.1, see the *OpenVMS Version 7.1 New Features Manual*.

5.1 Backup API

This section contains release notes pertaining to the Backup application programming interface (API). See Section 4.3 for additional notes about the Backup utility.

5.1.1 Problems and Restrictions

This section describes known problems and restrictions for the Backup API.

5.1.1.1 CONTROL Event Types Return Incorrect Message Arguments

V7.1

The Backup API returns incorrect message vector arguments to an event callback routine that handles event types BCK_EVENT_K_CONTROL and BCK_EVENT_K_STATISTICS. This means that the message vector, as passed to the application interface, is not suitable for output using the SYS$PUTMSG service.

For certain condition values, the application can correct the message vector. The condition value is located in the second longword of the message vector. The condition values and the required corrections are listed in the following table:

	Change First Word of Message Vector		Change Fifth Word of Message Vector	
Condition Value	**From**	**To**	**From**	**To**
BACKUP$_CNTRL_CONFCOPY	1	4	1	2
BACKUP$_CNTRL_CONFCOMP	1	4	1	2
BACKUP$_STAT_COMPARE	1	6		
BACKUP$_STAT_INACTIVE	1	5		
BACKUP$_STAT_PHYSICAL	1	7		
BACKUP$_STAT_RESTORE	1	6		
BACKUP$_STAT_SAVCOP_ACT	1	6		

5.1.1.2 Journaling Callback Events Restriction

V7.1

If an application registers a callback routine for any of the journaling events, it must register a callback routine for all the journalling callback events. The following is a list of the journaling callback events:

BCK_EVENT_K_JOURNAL_OPEN
BCK_EVENT_K_JOURNAL_WRITE
BCK_EVENT_K_JOURNAL_CLOSE

This is a permanent restriction. The documentation will be amended to incorporate this restriction.

See the Backup API chapter in the *OpenVMS Utility Routines Manual* for more information on registering callback routines.

5.1.1.3 Repetitive Calls to BACKUP$START Can Cause an Error

V7.1

Repetitive calls to BACKUP$START can cause the following error:

```
%BACKUP-F-INSBUFSPACE, insufficient buffer space
```

The number of repetitive calls completed before receiving this error varies, depending upon the previous backup operations performed.

The workaround for an application that receives this error is to exit the operation and restart.

This problem will be corrected in a future release.

5.1.2 Documentation Changes and Corrections

The following note contains a correction to the *OpenVMS Utility Routines Manual*.

5.1.2.1 *OpenVMS Utility Routines Manual*

V7.1

This section describes corrections to Chapter 3, Backup Routine, of the *OpenVMS Utility Routines Manual*.

In Chapter 3, the Description section for the Backup Routine contains an error in the first paragraph under the heading **BACKUP Event Callbacks**. The third sentence in that paragraph, beginning "To do so," should read:

"To do so, the application registers the callback routine by including option structure BCK_OPT_K_EVENT_CALLBACK in the call to BACKUP$START."

The manual will be corrected in a future release.

5.2 Batch and Print Queues

This section contains release notes pertaining to batch and print queues.

5.2.1 Problems and Restrictions

This section describes problems and restrictions pertaining to batch and print queues.

5.2.1.1 Terminating Executing Batch Jobs

V6.2

Under the following conditions, the DELETE/ENTRY command might fail to stop an executing batch job:

- The batch job is a DCL command procedure.
- There is an ON ERROR CONTINUE command (or SET NOON command) within the command procedure.

The DELETE/ENTRY command causes the job to terminate in phases. A delete_process AST routine is given in user mode, supervisor mode, and then executive mode. Because there is a small delay between each mode, it is possible that, in a batch job, a user-mode image might terminate and the command procedure might continue to execute until the supervisor-mode delete_process AST routine is executed.

The return status of the SYNCHRONIZE command is assumed to contain the termination status of the target batch job. In addition, command procedures would normally execute a command such as $ON ERROR THEN CONTINUE or $SET NOON before issuing the SYNCHRONIZE command. If a DELETE /ENTRY command is issued to the job executing the SYNCHRONIZE command, the JBC$_JOBABORT is interpreted as being the termination status of the target batch job rather than a return status of the SYNCHRONIZE command. The command procedure then continues to execute for a short period with this incorrect assumption and performs an operation such as requeuing the target batch job or incorrectly reporting a failure of the target batch job.

This problem has been fixed for the SYNCHRONIZE command by detecting this situation and waiting in an exit handler for longer than the delay between the user and supervisor mode termination delay.

Any other images that would report the job completion status obtained by the SJC$_SYNCHRONIZE_JOB function code of the $SNDJBC system service as the return status of the program should implement logic similar to the following:

1. Declare an exit handler
2. In the exit handler, implement the following logic:

```
IF (exit status is JBC$_JOBABORT)
THEN
    Wait 10 seconds
ENDIF
```

5.3 Debugger

This section contains release notes pertaining to the OpenVMS Debugger. Debugger release notes specifically about system management are in Section 4.4.

Unless specified otherwise, the release notes apply to both the character-cell and DECwindows Motif interfaces of the debugger.

5.3.1 Corrections

This section describes corrections to problems that existed in OpenVMS Debugger Version 7.0.

5.3.1.1 Breakpoints Within Fortran Routines

V7.1

In previous versions, when using the DECwindows Motif interface to debug Fortran programs, the debugger erroneously allowed users to set breakpoints with breakpoint toggles for subroutine entry masks, which do not contain executable code. This behavior has been corrected.

5.3.1.2 SET TYPE/OVERRIDE No Longer Ignored in Conditional Statements

V7.1

In previous versions, the debugger ignored the type setting specified by a SET TYPE/OVERRIDE command and returned a %DEBUG-E-OPNOTALLOW error when evaluating a conditional expression. For example, if *line* and *s1* were not of the same type, the following commands would result in an error message:

```
DBG> SET TYPE/OVERRIDE/BYTE
DBG> SET BREAK %LINE 12 WHEN (line[1] = s1)
```

This behavior has been corrected.

5.3.1.3 SET TRACE No Longer Restricted to One AST Level

V7.1

In previous versions, you could use the SET TRACE command to trace lines, instructions, branches, or calls of the mainline routines or of specified AST routines, but not both (at the same time).

This problem has been corrected. However, if the trace is established in mainline code, you must specify additional permanent tracepoints or breakpoints for each AST routine of interest. For example:

```
DBG> SET TRACE/LINE/INTO/NOSYS/NOSHARE
DBG> SET TRACE address-expression[,...]
```

where the first SET TRACE command is issued while suspended in mainline code, and *address-expression* is the address of an AST routine to be traced. Whenever the eventpoint suspends execution at the AST routine, the SET TRACE/LINE... command will take effect.

5.3.1.4 Quoted String Literals Now Passed Correctly

V7.1

In previous versions, the debugger erroneously did not allow passing a quoted string literal to a command procedure as a value or address type. This behavior has been corrected.

5.3.1.5 EXAMINE Command No Longer Fails on Valid Address Range

V7.1

In previous versions, when the language was set to C, an EXAMINE command of a valid range of addresses would sometimes fail with the message "%DEBUG-E-EXARANGE, invalid range of addresses." This behavior has been corrected.

5.3.1.6 Watchpoint Support in Global Sections (Alpha Only)

V7.1

In previous versions, watchpoints set on variables whose addresses are in global sections did not work. Attempting to set a watchpoint on a location in a global section resulted in a %DEBUG-E-BADWATCH message. This problem has been corrected on Alpha systems, but still exists on VAX systems (see the *OpenVMS Debugger Manual*).

5.3.1.7 Watchpoint Support and $GETJPI

V7.1

In previous versions, an active static watchpoint could cause changes in program behavior when both of the following conditions existed:

- The program made a system service call ($GETJPI) supported by SSI.
- As a result of that system call, the program branched based on information received about whether or not ASTs are enabled.

This behavior has been corrected, and the program receives correct information from $GETJPI about AST enablement.

5.3.1.8 DEBUG.EXE Image

V7.1

In previous versions of OpenVMS, you could not debug an image named DEBUG.EXE. This problem has been corrected on both VAX and Alpha systems.

5.3.2 Documentation Changes and Corrections

The following release note describes a minor addition to the *OpenVMS Debugger Manual*.

5.3.2.1 *OpenVMS Debugger Manual*

In Section 4.4, the list that begins with "On Alpha processors:" should contain the following additional bulleted item:

- You cannot deposit a value into register R30.

5.4 DEC Ada Run-Time Library

This section contains release notes pertaining to the DEC Ada Run-Time Library.

5.4.1 Problems and Restrictions

This section describes several potential problems.

5.4.1.1 AST Procedures With Access Violations

V7.0

DEC Ada written AST procedures can get access violations if the AST that causes their invocation occurs when the null thread or a non DEC Ada thread is running. A workaround exists if the procedure executes at the user level rather than the AST level. Instead of using a procedure, rewrite the program to use a task entry point that has pragma AST_ENTRY on it.

For more information about AST_ENTRY, refer to the section "Task Entries and OpenVMS Asynchronous System Traps" and the documentation on pragma AST_ENTRY in the *DEC Ada Language Reference Manual*.

5.4.1.2 Unexpected Storage Errors (Alpha Only)

V7.0

In OpenVMS Alpha Version 7.0 and later, binary compatibility fails for some DEC Ada programs that make incorrect assumptions about the amount of task space used by DEC Ada tasks. If a task gets a storage error that it did not previously get, you may need to add a length clause specifying the storage size for the task. If you already use a length clause, you might need to increase the amount of storage specified. This is necessary only in cases where the specified size (or default size) is not large enough for the task's execution.

5.5 DEC C Run-Time Library

The following sections contain release notes pertaining to the DEC C Run-Time Library (RTL).

5.5.1 Changes and Enhancements

This section describes changes and enhancements that are included in Version 5.2 or later of the DEC C RTL software. For more details, see the revision of the *DEC C Run-Time Library Reference Manual for OpenVMS Systems* that ships with DEC C Version 5.2 or later.

5.5.1.1 New Universal Symbols in DECC$SHR Improve Link Profile (Alpha Only)

V7.1

In OpenVMS Alpha Version 7.0, the DEC C Run-Time Library functions whose names are of the form decc$fmath_2, are defined in STARLET, but are not universal symbols in the DECC$SHR image. When these functions are referenced by an application, linking the image results in inclusion of the DECC$SHR image from IMAGELIB and the specific C math modules from STARLET. If neither the DPML$SHR image or the CMA$TIS_SHR image were already brought in during the IMAGELIB phase, the object form is included from STARLET because references to these symbols appear in the STARLET phase. In OpenVMS Version 7.0, this resulted in the undefined symbols:

```
CMA$TIS_ERRNO_SET_VALUE
CMA$TIS_VMS_ERRNO_SET_VALUE
```

In OpenVMS Alpha Version 7.1, the decc$fmath_2 symbols have been added to the DECC$SHR image so that both the DPML$SHR and CMA$TIS_SHR references are now resolved using shareable images found in IMAGELIB.

5.5.1.2 Time Zone Cache Greatly Improves Performance

V7.1

The UTC-based time functions, introduced in OpenVMS Version 7.0, had degraded performance compared with the non-UTC-based time functions. A cache for time zone files has been introduced to improve performance. The size of the cache is determined by the logical name DECC$TZ_CACHE_SIZE. To accommodate most countries changing the time twice per year, the default cache size is large enough to hold two time zone files.

5.5.1.3 Changing Default LRL Value On Stream Files

V7.1

In OpenVMS Version 7.0 the default LRL value on stream files was changed from 0 to 32767. This change caused significant performance degradation on certain file operations such as sort.

This is no longer a problem. The DEC C Run-Time Library now allows users to define the logical DECC$DEFAULT_LRL to change the default record-length value on stream files.

The DEC C Run-Time Library first looks for this logical. If it is found and it translates to a numeric value between 0 and 32767, that value is used for the default LRL.

To restore the behavior prior to OpenVMS Version 7.0, enter the following command:

```
$ DEFINE DECC$DEFAULT_LRL 0
```

5.5.1.4 Unicode Additions

V7.1

In OpenVMS Version 7.1, the DEC C Run-Time Library has added support for Unicode character set conversions. This release also includes converters that perform conversions between Unicode and any other supported character sets. (See Section 10.6 in the *DEC C Run-Time Library Reference Manual for OpenVMS Systems* for more information about character set conversions and a list of the supported character sets.)

The expanded set of converters includes converters for UCS-2, UCS-4, and UTF-8 Unicode encoding. The Unicode converters can be used by the ICONV CONVERT utility and by the iconv family of functions in the DEC C Run-Time Library.

5.5.1.5 Internationalization Support

V7.1

The DEC C RTL has added capabilities to allow application developers to create international software. The DEC C RTL obtains information about a language and a culture by reading this information from *locale* files.

If you are using these DEC C RTL capabilities, you must install a separate kit to provide these files to your system. The save set, VMSI18N071, is provided on the same media as the OpenVMS Version 7.1 operating system.

To install this save set, follow the standard OpenVMS installation procedures using this save set name as the name of the kit. There are several categories of locales that you can select to install. You can select as many locales as you need by answering the following prompts:

```
Do you want European and US support?
Do you want Chinese support?
Do you want Japanese support?
Do you want Korean support?
Do you want Thai support?
Do you want the Unicode converters?
```

This kit also has an Installation Verification Procedure that Digital recommends you run to verify the correct installation of the kit.

5.5.2 Problems and Restrictions

This section describes problems or restrictions related to using the DEC C RTL software.

5.5.2.1 Internationalization Compatibility Problem With DECwindows Motif

V7.1

Applications that call the Xlib locale routines in DECwindows Motif Version 1.2–3 using the method described in Section 4.18.5.3 of the *DECwindows Motif Version 1.2–3 for OpenVMS Release Notes* will continue to function on OpenVMS Version 6.2 and later. However, because the locale support in Xlib is not compatible with the support in the DEC C Run-Time Library (DECC$SHR.EXE), Xlib does not use the locale environment provided by the C library. Therefore, setting the locale in the C library does not affect the behavior of DECwindows Motif, although it does affect C library routines such as strcoll( ). Setting the locale in Xlib changes the behavior of DECwindows Motif but does not affect C library routines.

This problem is corrected in DECwindows Motif Version 1.2-4, and in the remedial kits for DECwindows Motif Version 1.2-3 that are described in Section 2.9.2.2.

5.5.2.2 Socket Behavior Prior to UCX Version 3.3

V7.0

In programs that link with a version of the DEC C RTL that supports passing sockets from a parent process to a child process and that links with DEC TCP/IP Services for OpenVMS (UCX) Versions 3.2 or earlier, a parent process cannot shut down or close a socket that has been closed by a child process, either specifically by the child or by default when the child terminates. With these versions of UCX, closing a socket breaks the TCP connection and another process cannot send any more data to it. For the `shutdown` function, the error status returned is EINVAL "invalid argument." For the `close` function, errno indicates EVMSERR.

Additionally, when a parent closes a socket, the socket is no longer available to any child processes that have inherited the socket in programs linked as specified above. On UCX Version 3.2 or earlier, a parent's closure of a socket interrupts operations pending from a child process on the same socket. In this case, a parent process should await termination of all its children before closing its sockets, and then ignore any errors when it does close them.

5.5.2.3 Linking DEC C Applications Using /NOSYSSHR

V7.0

When linking DEC C programs against the DEC C RTL object libraries using the /NOSYSSHR qualifier, applications that previously linked without undefined globals may now result in undefined globals for the CMA$TIS symbols. To resolve these undefined globals, add the following line to your link options file:

```
SYS$LIBRARY:STARLET.OLB/LIBRARY/INCLUDE=CMA$TIS
```

5.5.2.4 Linking With /NOSYSSHR Disables Dynamic Activation

V7.0

If a program linked with the /NOSYSSHR qualifier makes a call to a routine that resides in a dynamically activated image, the routine returns a value indicating an unsuccessful status, errno is set to ENOSYS, and vaxc$errno is set to C$_NOSYSSHR. The error message corresponding to C$_NOSYSSHR is "Linking /NOSYSSHR disables dynamic image activation." An example of this situation is a program linked with /NOSYSSHR that makes a call to a socket routine.

5.5.3 Documentation Changes and Corrections

This section describes corrections to the DEC C RTL documentation.

5.5.3.1 *DEC C Run-Time Library Reference Manual for OpenVMS Systems*

V7.1

The *DEC C Run-Time Library Reference Manual for OpenVMS Systems* erroneously states that the pipe function ignores the second parameter. This function has been corrected to ignore all values except O_NDELAY and O_NONBLOCK. The documentation for this second argument should be changed as follows:

flag

An optional argument used as a bitmask. If either the O_NDELAY or O_NONBLOCK bit is set, the I/O operations to the mailbox using array_fdscptr file descriptors terminate immediately, rather than waiting for another process.

If, for example, the O_NDELAY bit is set and the child issues a read request to the mailbox before the parent has put any data into it, the read terminates immediately with zero status. If neither the O_NDELAY nor O_NONBLOCK bit is set, the child waits on the read until the parent writes any data into the mailbox. This is the default behavior if no flag argument is specified.

The values of O_NDELAY and O_NONBLOCK are defined in the <fcntl.h> header file. Any other bits in the flag argument are ignored. You must specify this argument if the second optional, positional argument bufsize is specified. If the flag argument is needed only to allow specification of the bufsize argument, specify the flag as zero.

5.6 DECthreads

The following sections contain release notes pertaining to DECthreads. Release notes about kernel threads are in Section 5.10.

5.6.1 Changes and Enhancements

This section summarizes some significant changes and enhancements made to DECthreads. For detailed information about using DECthreads, refer to the *Guide to DECthreads*.

5.6.1.1 Behavior With Multiple Kernel Threads and Upcalls Enabled (Alpha Only)

V7.1

Starting with OpenVMS Alpha Version 7.1, DECthreads can use the kernel threads features introduced in OpenVMS Alpha V7.0, provided these features have been enabled in a multithreaded process.

The following sections describe changes in DECthreads behavior when multiple kernel threads and upcalls are enabled.

By default, the kernel threads features are disabled for an application. Refer to Section 5.10.1 for information on enabling kernel threads for a multithreaded process.

5.6.1.1.1 Multiprocessor Support

Starting with OpenVMS Alpha Version 7.1, DECthreads can create virtual processors as they are needed by the application. The number of virtual processors that DECthreads can create is limited by the system parameter MULTITHREAD. Typically, this parameter is set to the number of processors that reside in the system. Regardless of this parameter's value, DECthreads will create no more virtual processors than there are user threads (excluding DECthreads internal threads).

DECthreads does not delete virtual processors or let them terminate. They are retained in an idle (HIB) state until they are needed again. During image rundown, they are deleted by the OpenVMS executive.

The DECthreads scheduler can schedule any user thread onto any virtual processor. Therefore, a user thread can run on different kernel threads at different times. Normally, this should pose no problem. However, you may notice small changes; for example, a user thread's PID (retrieved by querying the system) might change from time to time.

See Section 5.10.1 for more information about kernel threads.

5.6.1.1.2 $EXIT and Exit Handling Changes

With OpenVMS Alpha Version 7.1, $EXIT and exit handling for multithreaded processes changes significantly when kernel threads is enabled. Exit handlers are now executed in a separate thread, not in the thread that calls the $EXIT system service routine. A call to $EXIT in a multithreaded process results in a call to `pthread_exit`. See the *Guide to DECthreads* for more information.

5.6.1.1.3 Condition Variable Waits

On OpenVMS Alpha Version 7.1, threaded applications that use the condition wait and timed condition wait routines may experience more spurious wakeups than previously. Correctly coded applications should not be affected by this change. See the *Guide to DECthreads* for more information.

5.6.1.1.4 Blocking System Service Calls

Starting with OpenVMS Alpha Version 7.1, system service calls that block are now thread synchronous; that is, they block only the calling thread instead of the entire process. See the *Guide to DECthreads* for more information.

5.6.1.1.5 $HIBER

Starting with OpenVMS Alpha Version 7.1, threaded applications that use $HIBER may experience more spurious wakeups than in previous releases. After the call to $HIBER returns, you must validate the wakeup to ensure that it is appropriate for execution to proceed. This wakeup validation has always been required, but it is now more important. See the *Guide to DECthreads* for more information.

Prior to Version 7.1, a thread that called $HIBER (or that called a library routine that resulted in a call to $HIBER) would cause the whole process to hibernate briefly whenever that thread was scheduled to run. With multiple threads in simultaneous calls to $HIBER, there was no reliable way to wake one or all of the threads. The next hibernating thread to be scheduled would awaken, and any other threads would continue to sleep.

With Version 7.1, these problems are resolved. However, the new behavior has other effects. For instance, hibernation-based services such as LIB$WAIT and the C RTL `sleep( )` routine might complete prematurely if the service does not validate its wakeup (to ensure that enough time has passed or that there is some other reason for it to return).

5.6.1.2 POSIX 1003.1c Standard Interface

V7.0

Starting with OpenVMS Version 7.0, the DECthreads library (PTHREAD$RTL.EXE) implements a POSIX 1003.1c standard interface. The new POSIX.1c (or "pthread") interface is the most portable, efficient, and powerful OpenVMS programming interface for a multithreaded environment. This interface is defined by the C language header file pthread.h.

No exception-returning interface exists for the POSIX 1003.1c standard interface.

5.6.1.3 Thread Independent Services (TIS) Interface

V7.0

OpenVMS Version 7.0 and later includes the Thread Independent Services (TIS) application programming interface (CMA$TIS_SHR.EXE). TIS provides services that assist with the development of thread-safe application programming interfaces (APIs).

Thread synchronization can involve significant run-time cost, which is undesirable in the absence of threads. TIS enables you to build thread-safe APIs that are efficient in the nonthreaded environment, yet provide the necessary synchronization in the threaded environment.

When DECthreads is not active within the process, TIS executes only the minimum steps necessary. Code running in a nonthreaded environment is not burdened by the run-time synchronization that is necessary when the same code is run in a threaded environment. When DECthreads is active, the TIS functions provide the necessary thread-safe synchronization.

5.6.1.4 POSIX 1003.4a Draft 4 Interface Retirement

V7.0

The POSIX 1003.4a, Draft 4 (or "d4") interface of DECthreads is slated for retirement in a future release. Applications that were written using the POSIX 1003.4a, Draft 4 interface should be migrated to the new POSIX 1003.1c standard (or "pthread") interface provided by DECthreads. A compatibility mode for the Draft 4 POSIX 1003.4a interface has been provided in this release to help ease migration. This compatibility mode will be removed in a future release.

5.6.1.5 CMA Interface: Change in Status

V7.0

In future releases, the DECthreads CMA interface (or "cma") will continue to exist and be supported in the OpenVMS operating system. However, starting with this release, it will no longer be documented or enhanced. Therefore, Digital recommends that you port your CMA-based application to the IEEE POSIX 1003.1c-1995 standard (or "pthread") interface provided by DECthreads.

5.6.1.6 Application Coding Errors

V7.0

Substantial changes made to threads in OpenVMS Version 7.0 will likely expose programming errors in existing applications that use DECthreads. Such errors include, but are not limited to, the following:

- Attempting to unlock a mutex that is not locked
- Use of uninitialized variables (for example, uninitialized condition variables)
- Improper use of data structures (for example, using a `pthread_attr_t` instead of a `pthread_mutexattr_t` in a call to `pthread_mutexattr_create( )`)
- Improper data access synchronization
- Use of an undocumented or unsupported routine

5.6.2 Problems and Restrictions

This section describes known DECthreads problems and restrictions.

5.6.2.1 Release Compatibility

V7.0

Binary compatibility and a source compatibility mode for the Draft 4 POSIX interface and the CMA interface to DECthreads are provided starting with OpenVMS Version 7.0. However, no object compatibility for object modules created on earlier OpenVMS releases is provided for object modules that use the Draft 4 POSIX interface.

5.6.2.2 Language Support

V7.0

This release does not include interface definitions for non-C languages for the POSIX 1003.1c standard style interface to DECthreads. All DECthreads routines are usable from non-C languages; however, the application must provide the routine declarations. These self-defined declarations should be modeled after the C language declarations in pthread.h.

5.6.2.3 DECthreads Debugger Metering Function

V7.0

The metering capability of the DECthreads debugger does not work in this release.

5.6.2.4 C Run-Time Library *errno* Value

V7.0

When *errno* is accessed from the OpenVMS Debugger, the value of the global errno (not the per-thread errno) is returned. (This is not a new condition; it just has not been documented previously.)

5.6.2.5 SET TASK/ACTIVE Command

V6.2

The OpenVMS Debugger command SET TASK/ACTIVE does not work for DECthreads (on both OpenVMS Alpha and VAX systems) or for DEC Ada for OpenVMS Alpha systems, the tasking for which is implemented using DECthreads.

Instead, you can use the following effective substitutes on DECthreads:

- For query-type actions, use the SET TASK/VISIBLE command.
- To gain control to step through a particular thread, use strategic placement of breakpoints.

5.6.2.6 Dynamic Image Activation

V6.2

Applications that use thread-safe run-time libraries might not be able to use LIB$FIND_IMAGE_SYMBOL to dynamically activate DECthreads or products that depend on DECthreads.

Certain run-time libraries use conditional synchronization mechanisms. These mechanisms typically are enabled during image initialization when the run-time library is loaded only if the process is multithreaded. If the process is not multithreaded, the synchronization is disabled.

If the application dynamically activates DECthreads, any run-time library using conditional synchronization may not behave reliably.

To work around this problem, link the image that calls LIB$FIND_IMAGE_SYMBOL against PTHREAD$RTL.

5.7 DECTPU for DECwindows Motif

The following sections contain release notes pertaining to DECTPU for DECwindows Motif.

5.7.1 Problems and Restrictions

This section describes DECTPU for DECwindows Motif problems and restrictions.

5.7.1.1 Small Display Monitors and DECwindows Motif Applications

V6.0

When running DECwindows Motif DECTPU on small display monitors, the main window can be less than fully visible.

To correct this condition, follow these steps:

1. Add the following resources to your DECTPU X resource file:

```
Tpu.Tpu$MainWindow.X:                    0
Tpu.Tpu$MainWindow.Y:                    0
Tpu.Tpu$Mainwindow.Rows:                 21
Tpu*condensedFont:                       on
Tpu*fontSetSelection:                    1
```

2. Copy the resource file from `SYS$LIBRARY:EVE.DAT` and add the previous lines.
3. Use logical name TPU$DEFAULTS to point at the new resource file.

   The following example invokes the EVE DECwindows Motif user interface using the X resource file named eve_small_window.dat in your login directory to edit the file LOGIN.COM.

```
$ DEFINE TPU$DEFAULTS SYS$LOGIN:EVE_SMALL_WINDOW.DAT
$ EDIT/TPU/INTER=DECWINDOWS LOGIN.COM
```

5.8 Digital Portable Mathematics Library (DPML) (Alpha Only)

This section contains release notes pertaining to the Digital Portable Mathematics Library (DPML). See the *Digital Portable Mathematics Library* for information about using this utility.

5.8.1 Changes and Enhancements

This section describes a change to values returned in some DPML routines.

5.8.1.1 Returned Values Are Different in Some DPML Routines

V7.1

For OpenVMS Alpha Version 7.1, many of the DPML routines have been modified or rewritten to improve their performance or accuracy.

Occasionally these changes cause the modified routines to return values that differ by one least significant bit (lsb) when compared to results obtained in previous versions.

The results of the routines in this release are generally more accurate than the results in the previous release. This difference in results should not be considered a bug in the new version of the library.

5.9 High-Performance Sort/Merge Utility (Alpha Only)

For programming release notes pertaining to the callable interface (SOR routines) of the OpenVMS Alpha high-performance Sort/Merge utility, see Section 3.3.

For information about using the OpenVMS Alpha high-performance Sort/Merge utility, refer to the *OpenVMS Utility Routines Manual* and the *OpenVMS User's Manual*.

5.10 Kernel Threads (Alpha Only)

This section contains release notes pertaining to using kernel threads on OpenVMS Alpha systems. Release notes about DECthreads are in Section 5.6.

5.10.1 Changes and Enhancements

V7.1

Several changes have been made to kernel threads in OpenVMS Alpha Version 7.1 to let you control whether kernel threads is enabled for a process.

You can safely use an application with kernel threads if the threads do not interfere with each other during their independent execution. However, if they do interfere, you may not wish to enable kernel threading.

One example of interference is when two or more threads modify a common memory location without synchronization. This condition can lead to incorrect and inconsistent results. Using kernel threading, where the threads run concurrently on multiple CPUs, can aggravate this problem. With multiple kernel threading disabled for a process, an application runs much less risk of encountering this problem.

The following changes have been made to kernel threads in OpenVMS Alpha Version 7.1 to give you more control over enabling or disabling kernel threads:

- MULTITHREAD System Parameter

  In OpenVMS Alpha Version 7.0, the default behavior for applications that use DECthreads was to run without the kernel threads features introduced in that release. Kernel threads was disabled systemwide by setting the MULTITHREAD system parameter to 0.

  In OpenVMS Alpha Version 7.1, AUTOGEN now sets the MULTITHREAD system parameter value equal to the number of CPUs in the system. This setting makes the kernel threads features available systemwide for applications that choose to enable them. For more information about the MULTITHREAD system parameter, see the *OpenVMS System Management Utilities Reference Manual*, or SYSMAN or SYSGEN help.

- /THREADS_ENABLE Linker Qualifier and THREADCP Tool

  Once kernel threads features are enabled systemwide, the determination of whether an application uses these features is made at the main image level. Information stored in the image header of a main image dictates whether the features are to be used. The features are disabled by default.

  When building new images, you can control this information by using the new linker qualifier /THREADS_ENABLE. (For details about the /THREADS_ENABLE qualifier, see the *OpenVMS Linker Utility Manual* or DCL help for the LINK command.)

  To modify the control information in existing images, use the THREADCP tool described in the *OpenVMS Version 7.1 New Features Manual*. The THREADCP tool enables testing to verify that a threaded application runs correctly with the new kernel threads features.

For more information about kernel threads, refer to the *OpenVMS Alpha Guide to Upgrading Privileged-Code Applications*.

5.11 Librarian Utility

The following sections contain release notes pertaining to the Librarian utility (LIBRARIAN).

5.11.1 Problems and Restrictions

This section describes known LIBRARIAN problems and restrictions.

5.11.1.1 PGFLQUOTA Should Exceed 23000 (Alpha Only)

V1.5

The OpenVMS Alpha LIBRARIAN sometimes does not inform you of errors during compression, data reduction, or data expansion operations. This problem occurs if the account or process in which the LIBRARIAN is running has a low PGFLQUOTA process quota. Operation failure is not readily apparent because the $PUTMSG system service always returns a status of SS$_NORMAL, even when the system service fails. However, when a failure occurs, the LIBRARIAN returns a status other than success.

To work around this problem, run the compression, data reduction, or data expansion operation in an account with a PGFLQUOTA process quota greater than 23000. In addition, ensure that your command procedures check the return status from the LIBRARY command.

5.12 LTDRIVER

The following sections contain release notes pertaining to the LTDRIVER.

5.12.1 Problems and Restrictions

This section describes known LTDRIVER problems and restrictions.

5.12.1.1 CANCEL SELECTIVE Cannot Cancel IO$_TTY_PORT Functions

V6.1

In releases prior to OpenVMS Version 6.1, LTDRIVER did not set the "extended DDT" bit; therefore, the POSIX function CANCEL SELECTIVE did not work with LTDRIVER. This has been fixed, but a restriction remains.

Although this fix allows $QIO reads and writes to be selectively canceled, any $QIO done to the port driver (that is, with the IO$_TTY_PORT function modifier—like a LAT connect $QIO) *cannot* be canceled with CANCEL SELECTIVE.

5.13 MACRO-32 Compiler for OpenVMS Alpha (Alpha Only)

The following sections contain release notes pertaining to the MACRO-32 Compiler for OpenVMS Alpha.

5.13.1 Problems and Restrictions

This section describes problems and restrictions pertaining to the MACRO-32 Compiler for OpenVMS Alpha.

5.13.1.1 Quadword Moves Into the VAX SP and PC

V7.1

Due to architectural differences between VAX and Alpha computers, it is not possible to completely emulate quadword moves into the VAX stack pointer (SP) and the program counter (PC) without programmer intervention. The VAX architecture defines R14 as the SP and R15 as the PC. A MOVQ instruction with SP as the target would simultaneously load the VAX SP and PC, as shown in the following example:

```
MOVQ    R0,SP   ; Contents of R0 to SP, R1 to PC

MOVQ    G^CTL$AQ_STACK+16, SP   ; ctl$aq_stack+16 to SP
                                ; ctl$aq_stack+20 to PC
```

Note

Even though the VAX MACRO assembler permits this operation, it is specifically disallowed because the results are unpredictable (see the *VAX Architecture Reference Manual*, second edition).

If the compiler encounters a MOVQ instruction with SP as the destination, it generates a sign-extended longword load from the supplied source into R30 (the Alpha stack pointer) and issues the following informational message:

```
%AMAC-I-CODGENINF, (1) Longword update of Alpha SP, PC untouched
```

If the intended use of the MOVQ instruction is to achieve the VAX behavior, the MOVQ should be followed by a branch to the intended address, as shown here:

```
MOVQ    G^CTL$AQ_STACK+16, SP   ; Load the SP
JMP     G^CTL$AQ_STACK+20       ; And branch
```

If the intended use of the MOVQ instruction is to load the stack pointer with an 8 byte value, the EVAX_LDQ built-in should be used instead, as follows:

```
EVAX_LDQ        G^CTL$AQ_STACK+16
```

5.13.1.2 .CALL_ENTRY Requires New STARLET.MLB

V7.0

The MACRO-32 compiler has been modified to include 64-bit addressing support. When you use this new version of the compiler, regardless of whether you use the 64-bit addressing features, you must also use the OpenVMS Version 7.0 (or later) ALPHA$LIBRARY:STARLET.MLB.

STARLET.MLB contains the MACRO_COMPILER_DIRECTIVES macro definitions, including the revised .CALL_ENTRY directive. If you do not use ALPHA$LIBRARY:STARLET.MLB from OpenVMS Version 7.0 or later, your program will not compile successfully. Instead, the following errors will be reported:

```
%AMAC-E-DIRARGERR, error in compiler directive argument
at line number n in file filename

%AMAC-E-DIRSYNX, directive syntax error
at line number n in file filename
```

Make sure that ALPHA$LIBRARY:STARLET.MLB from OpenVMS Version 7.0 or later is on your system and that the ALPHA$LIBRARY logical points to the correct directory.

If necessary, you can extract .CALL_ENTRY from the new STARLET.MLB, and insert it in an alternate STARLET, if you are using one.

5.14 Mail Utility

This section contains release notes about the Mail utility that are of interest to programmers. Release notes of interest to system managers are documented in Section 4.12.

5.14.1 Changes and Enhancements

The following section describes a change made to some MAIL utility routines.

5.14.1.1 MAIL$SEND, MAIL$MESSAGE, and MAIL$MAILFILE Return Status Modifications

V7.1

The MAIL utility routines MAIL$SEND, MAIL$MESSAGE, and MAIL$MAILFILE have been modified to return a word-sized length for items specified in the output item list.

5.14.2 Problems and Restrictions

The following note describes a restriction for callable mail.

5.14.2.1 Threads Restriction for Callable Mail

V7.1

OpenVMS callable mail routines are *not* thread-safe. Refer to the *Guide to DECthreads* for more information about calling nonthread-safe routines within a threaded application.

Because callable mail context information is maintained on a per-process (rather than a per-thread) basis, multiple threads performing context-based processing must be synchronized so that only *one* mail context of a given type is active at once. Otherwise one thread could corrupt another thread's mail operations.

On OpenVMS Alpha systems, there is an additional restriction when kernel threads is enabled in a multithreaded environment. In this environment, callable mail should be used only in the initial thread.

5.15 Mathematics (MTH$) Run-Time Library

The following sections contain release notes pertaining to the Run-Time Mathematics Library (MTH$).

5.15.1 Problems and Restrictions

This section describes known MTH$ problems and restrictions.

5.15.1.1 Linking Images to Run on Previous OpenVMS VAX Versions (VAX Only)

V6.1

This version of OpenVMS VAX provides updated versions of the Mathematics Run-Time Library (RTL) images MTHRTL.EXE, UVMTHRTL.EXE, and VMTHRTL.EXE that contain new entry points in support of DEC Fortran Version 6.0. (UVMTHRTL.EXE is an alternate form of MTHRTL.EXE; references to MTHRTL.EXE in the following paragraphs also apply to UVMTHRTL.EXE.)

Due to the large number of entry points added to MTHRTL.EXE, that image's transfer vector was extended and its global section match identifier incremented. This means that images linked against the new version of MTHRTL.EXE will not run on a system running a previous version of OpenVMS VAX, unless that system has also installed DEC Fortran Version 6.0. In addition, images linked against the new MTHRTL.EXE cannot be translated to run on OpenVMS Alpha using DECmigrate.

To link an image so that it will run on a previous version of OpenVMS VAX, create a directory that contains saved copies of the .EXE and .OLB files from the SYS$LIBRARY directory of the earliest version you wish to support, and define the logical name SYS$LIBRARY to point to that directory before linking. Because OpenVMS VAX also defines a system logical name MTHRTL to refer to either MTHRTL.EXE or UVMTHRTL.EXE, you must also define MTHRTL as a logical name in the process or job table to point to the copy in the directory of older images. For example:

```
$ DEFINE/USER SYS$LIBRARY disk:[OLD_SYSLIB]
$ DEFINE/USER MTHRTL SYS$LIBRARY:MTHRTL.EXE
$ LINK ...
```

Images to be translated using DECmigrate should be linked against the SYS$LIBRARY files of OpenVMS VAX Version 5.5-2 or earlier.

5.16 POLYCENTER Software Installation Utility

The release notes in this section pertain to the POLYCENTER Software Installation utility. Also see Section 4.16 for notes about this utility that are of interest to system managers.

5.16.1 Problems and Restrictions

This section describes known problems and restrictions with using the POLYCENTER Software Installation utility to create software kits. Problems and restrictions of interest to system managers are described in Section 4.16.2.

5.16.1.1 Generation Option of File Statement

V7.1

The generation option of the *file* statement does not work correctly on product removal.

For example, the product description files for products TEST1 and TEST2 are as follows:

```
product DEC VAXVMS TEST1 V1.0 full ;
    file [SYSEXE]TEST.EXE generation 1 ;
end product ;
```

```
product DEC VAXVMS TEST2 V1.0 full ;
    file [SYSEXE]TEST.EXE generation 2 ;
end product ;
```

Installing product TEST1 followed by product TEST2 works correctly. Generation 2 of file TEST.EXE replaces generation 1 of file TEST.EXE. However, if you remove product TEST2, generation 1 of file TEST.EXE is not restored; generation 2 of the file is left on the system.

One workaround is to use an *execute install...remove* statement sequence in product TEST2. The command procedure invoked during the *execute install* portion of the statement saves any previous version of the file. Later the *execute remove* portion of the statement restores the saved version of the file.

Digital expects to correct this problem in a future release.

5.16.1.2 Multiple Execute Remove Statements

V6.1

There is a problem with the *execute remove* statement where only the first one executes during a remove operation. However, all of the *execute install* statements execute during an install operation.

Digital expects to correct this problem in a future release.

5.17 Privileged Interfaces and Data Structures (Alpha Only)

V7.1

Privileged-code applications that were recompiled and relinked to run on OpenVMS Alpha Version 7.0 do not require source-code changes and do not have to be recompiled and relinked to run on OpenVMS Alpha Version 7.1.

Privileged-code applications from releases prior to OpenVMS Alpha Version 7.0 that were not recompiled and relinked for OpenVMS Alpha Version 7.0 must be recompiled and relinked to run on OpenVMS Alpha Version 7.1.

OpenVMS Alpha Version 7.0 included significant changes to OpenVMS Alpha privileged interfaces and data structures. As a result of these changes, privileged-code applications from releases prior to OpenVMS Alpha Version 7.0 might require source-code changes to run correctly on OpenVMS Alpha Version 7.0 and later. For more details about OpenVMS Alpha Version 7.0 changes that may require source changes to privileged-code applications, see the *OpenVMS Alpha Guide to Upgrading Privileged-Code Applications*.

For information about recompiling and relinking device drivers, see Chapter 6.

5.18 Record Management Services (RMS)

This section contains release notes pertaining to RMS.

5.18.1 Changes and Enhancements

The following note describes a change that affects RMS.

5.18.1.1 RMS Performance Optimization With VIOC Cache (Alpha Only)

V7.1

OpenVMS Alpha Version 7.1 enables RMS to use a feature of the VIOC cache that eliminates unnecessary RMS thread switches. During disk reads from Files–11 devices, RMS now avoids a thread switch if the read request is immediately satisfied from the VIOC cache.

This functionality increases the number of RMS operations that complete without stalling. Some RMS operations that always stalled previously now may never stall.

Applications might be affected by two changes in behavior:

- Some asynchronous RMS operations that used to return RMS$_PENDING, an alternate success, now return RMS$_NORMAL.

- Applications that request FAB/RAB$V_SYNCSTS on an asynchronous call might receive RMS$_SYNCH, an alternate success, on operations that never returned this status before. When RMS$_SYNCH is returned, the user's AST completion routine is not called. There is no change in user completion AST behavior for synchronous RMS calls or for asynchronous calls that do not set FAB/RAB$V_SYNCSTS.

If these changes in behavior are a problem for your application, you can execute the following command to disable VIOC caching for the affected files:

```
$ SET FILE/DATA_CHECK=READ filespec
```

5.18.2 Documentation Changes and Corrections

The following note contains a correction to the *OpenVMS Record Management Services Reference Manual*.

5.18.2.1 *OpenVMS Record Management Services Reference Manual*

V7.1

Section 6.1 of the *OpenVMS Record Management Services Reference Manual* incorrectly states that a maximum multibuffer count of 255 can be specified for the RAB$B_MBF field. The actual maximum multibuffer count allowed is 127. This maximum has never changed.

5.19 Run-Time Library (LIB$)

This section contains release notes pertaining to the Run-Time Library (LIB$).

5.19.1 Problems and Restrictions

This section describes known LIB$ RTL problems and restrictions.

5.19.1.1 LIB$FIND_IMAGE_SYMBOL Signals Warning for Modules With Compilation Errors

V7.1

LIB$FIND_IMAGE_SYMBOL may signal a warning (LIB$_EOMWARN) to indicate that the image being activated contains modules that had compilation warnings. A condition handler used in conjunction with LIB$FIND_IMAGE_SYMBOL should probably handle this as a special case.

To allow LIB$FIND_IMAGE_SYMBOL to continue execution after signalling LIB$_EOMWARN, the condition handler should exit with SS$_CONTINUE. For this reason, you may choose not to use LIB$SIG_TO_RET as a condition handler for LIB$FIND_IMAGE_SYMBOL.

5.20 Screen Management (SMG$) Facility

The following sections contain release notes pertaining to the Screen Management (SMG$) Facility.

5.20.1 Documentation Changes and Corrections

The following note describes several minor corrections to the *OpenVMS RTL Screen Management (SMG$) Manual*.

5.20.1.1 *OpenVMS RTL Screen Management (SMG$) Manual*

V7.1

Note the following information that should be added to topics in the reference section at the end of the *OpenVMS RTL Screen Management (SMG$) Manual*:

- In the documentation for the SMG$READ_COMPOSED_LINE routine, the following text should be appended to the description of the **flags** argument:

  "The terminal characteristic /LINE_EDITING should be set for your terminal in order for these flags to work as expected. /LINE_EDITING is the default."

- The description of routine SMG$SET_KEYPAD_MODE should contain this note:

  Note

  Changing the keypad mode changes the physical terminal setting. This is a global change for all virtual keyboards, not just the virtual keyboard specified by the **keyboard-id** argument.

5.21 Shared Linkage Sections

This section contains release notes pertaining to shared linkage sections.

5.21.1 Problems and Restrictions

This section describes a restriction on using libraries installed with shared linkage.

5.21.1.1 Interdependencies Between Images (Alpha Only)

V6.1

On OpenVMS Alpha systems, if you want to use an alternate version of any library installed with shareable linkage, you must use alternate (noninstalled) versions of all the libraries that call that library. The libraries that can be installed with shared linkage are LIBOTS, LIBRTL, CMA$TIS_SHR, DPML$SHR, and DECC$SHR.

The dependencies are in the order listed.

For example, if you issue the command:

```
$ DEFINE DPML$SHR SYS$LIBRARY:DPML$SHR.EXE;
```

then you must also issue the following command:

```
$ DEFINE LIBOTS SYS$LIBRARY:LIBOTS.EXE;
```

Failure to redefine all calling libraries may result in access violations.

5.22 System Services

The following sections contain release notes pertaining to system services.

All system services are documented in the *OpenVMS System Services Reference Manual*.

5.22.1 Problems and Restrictions

This section describes problems and restrictions related to system services.

5.22.1.1 SYS$AVOID_PREEMPT and SYS$SETUP_AVOID_PREEMPT Prototype Definitions

V7.1

The new system services SYS$AVOID_PREEMPT and SYS$SETUP_AVOID_PREEMPT were inadvertently omitted from the system service prototype definitions. Consequently, the compile-time definitions for these services are not included in STARLET.H for C or in macro definitions for other languages.

For this release, you must manually create the prototype definitions. Each service takes one integer parameter, ENABLE, which is 1 to enable and 0 to disable. For example, the C prototypes are defined as follows:

```
int sys$avoid_preempt(int enable);
int sys$setup_avoid_preempt(int enable);
```

Other languages should have equivalent definitions.

The prototypes for these services will be included in the next release. Note that the link-time definitions for these services are included correctly in Version 7.1.

5.22.1.2 Linking SECURESHR Images to Run on Older Versions

V7.0

Some additional entry points have been added to the shareable image dispatch vector. Because of this change, any applications linked against Version 7.0 or later of SECURESHR will not run on a pre-Version 7.0 system. System services that use SECURESHR are the following:

$FORMAT_ACL
$PARSE_ACL
$FORMAT_AUDIT
$DELETE_INTRUSION
$SCAN_INTRUSION
$SHOW_INTRUSION
$ADD_PROXY
$DELETE_PROXY
$DISPLAY_PROXY
$VERIFY_PROXY

If your program uses any of these system services and you want to create a version that runs on systems prior to Version 7.0, you must link your program on a system running a version of OpenVMS prior to Version 7.0.

5.22.1.3 $SUSPND Behaves Incorrectly in a Cluster Environment

VAX V6.0
Alpha V1.5

When the $SUSPND system service is called and the target process is on a different cluster node than that of the process calling the $SUSPND service, the kernel mode suspend flag (bit 0) is ignored. As a result, any suspend is treated as a supervisor-mode suspend.

5.22.2 Documentation Changes and Corrections

The notes in this section describe changes and corrections to the *OpenVMS System Services Reference Manual*.

5.22.2.1 *OpenVMS System Services Reference Manual*

V7.1

In the description of the $CRMPSC_GDZRO_64 system service, the format section erroneously cites the SYS$CRMPSC_GPFILE_64 system service. This reference to SYS$CRMPSC_GPFILE_64 should be replaced with $CRMPSC_GDZRO_64. The argument list is correct.

5.23 X/Open Transport Interface (XTI)

The notes in this section describe the X/Open Transport Interface (XTI).

5.23.1 Changes and Enhancements

OpenVMS Version 6.2 and later supports the X/Open Transport Interface (XTI) programming interface. The implementation conforms with the XPG4 X/Open CAE XO/CAE/91/600 (ISBN 1 872630 29 4) X/Open Transport Interface (XTI) specification.

Supported Transports

OpenVMS Version 7.1 supports the DECnet for OpenVMS (Phase IV), DECnet-Plus for OpenVMS (Phase V), and TCP/IP transports. See Section 5.23.2 for support restrictions.

The transport names used in the t_open routine are 'dnet' for DECnet and either 'ip/udp' or 'ip/tcp' for TCP/IP.

Other transports are available with other networking layered products. Consult individual layered product documentation for information about OpenVMS XTI support.

Architecture

XTI is supported by front end and back end code. Front end code provides access to the standard interface routines. Back end code provides the interface from the front end code to the selected networking transport.

The supporting image files are as follows:

XTI front end code	SYS$SHARE:XTI$XTILIB.EXE
TCP/IP XTI back end code	SYS$SHARE:XTI$IPSHR.EXE
DECnet XTI back end code	SYS$SHARE:XTI$DNETSHR.EXE
XTI C programming include file	SYS$LIBRARY:XTI.H

Linking Requirements

After compiling an XTI program, no additional qualifiers are required for linking with XTI.

Documentation

Documentation about XTI is not included in the OpenVMS documentation set. You can order documentation directly from X/Open Company Limited. If you have access to the Internet, you can get more information about X/Open Company Limited (including their publications) by browsing the following URL:

http://www.xopen.co.uk/

You can also contact X/Open Company Limited at the following locations:

- USA: East Coast

  X/Open Company Limited
  3141 Fairview Park Drive
  Falls Church
  VA 22042-4501
  U.S.A.
  Tel: +1 (703) 876 0044
  Fax: +1 (703) 876 0050

- USA: West Coast

  X/Open Company Limited
  1010 El Camino Real
  Suite 380
  Menlo Park, CA 94025
  U.S.A.
  Tel: +1 (415) 323 7992
  Fax: +1 (415) 323 8204

- United Kingdom:

  X/Open Company Limited
  Apex Plaza
  Forbury Road
  Reading
  Berks RG1 1AX
  U.K.
  Tel: +44 1734 508311
  Fax: +44 1734 500110

- Japan:

  X/Open Company Limited
  Karufuru-Kanda Bldg, 9F
  1-2-1 Kanda Suda-cho
  Chiyoda-Ku
  Tokyo 101
  Japan
  Tel: +81 3 3251 8321
  Fax: +81 3 3251 8376

5.23.2 Problems and Restrictions

V6.2

The following restrictions apply to XTI on OpenVMS systems:

- Nonblocking I/O is unsupported for DECnet for OpenVMS.

  DECnet for OpenVMS (Phase IV) does not fit the model for nonblocking I/O. Attempts to open or switch an XTI file descriptor to nonblocking I/O (O_NONBLOCK) will fail.

- Connectionless I/O is unsupported for DECnet for OpenVMS.

  DECnet for OpenVMS (Phase IV) does not fit the model of connectionless I/O. Therefore, only connection-oriented connections are supported.

- Disabled ASTs cause problems.

  The XTI back end code uses ASTs for asynchronous delivery of events from the transports. If ASTs are disabled (sys$setast(0)), the XTI back end code will not operate correctly until the ASTs are enabled again.

- XTI file descriptors are not compatible with C Run-Time Library file descriptors.

In addition, the 't_info' structure returned from the t_open function reports any additional implementation-specific restrictions for the given transport. (See the XTI documentation for information about the t_open command.)

6

Device Support on OpenVMS Systems

This chapter contains release notes pertaining to OpenVMS device support on Alpha and VAX systems. Where appropriate, section headings indicate whether specific notes contain Alpha-specific or VAX-specific information.

6.1 Recompiling and Relinking OpenVMS Device Drivers

The following sections contain release notes pertaining to recompiling and relinking OpenVMS device drivers.

6.1.1 Alpha and VAX SCSI Device Drivers

V7.1

All OpenVMS Alpha and OpenVMS VAX SCSI device drivers from OpenVMS Version 7.0 and earlier must be recompiled and relinked to run correctly on OpenVMS Version 7.1.

You must recompile and relink these drivers because OpenVMS Version 7.1 includes changes to the data structures used by OpenVMS Alpha and OpenVMS VAX SCSI device drivers. No source changes are required by these data structure changes.

If you have an Alpha SCSI driver and you are upgrading from a version prior to OpenVMS Alpha 7.0, see Section 6.1.2.

6.1.2 OpenVMS Alpha Device Drivers

V7.1

Device drivers that were recompiled and relinked to run on OpenVMS Alpha Version 7.0 do not require source-code changes and do not have to be recompiled and relinked to run on OpenVMS Alpha Version 7.1. (Note that Alpha SCSI drivers, however, must be recompiled and relinked as described in Section 6.1.1.)

Device drivers from releases prior to OpenVMS Alpha Version 7.0 that were not recompiled and relinked for OpenVMS Alpha Version 7.0 must be recompiled and relinked to run on OpenVMS Alpha Version 7.1.

OpenVMS Alpha Version 7.0 included significant changes to OpenVMS Alpha privileged interfaces and data structures. As a result of these changes, device drivers from releases prior to OpenVMS Alpha Version 7.0 might also require source-code changes to run correctly on OpenVMS Alpha Version 7.0 and later. For more details about OpenVMS Alpha Version 7.0 changes that may require source changes to customer-written drivers, see the *OpenVMS Alpha Guide to Upgrading Privileged-Code Applications*.

6.2 Changed Behavior of IO$_SKIPFILE Function

V7.1

The performance of the IO$_SKIPFILE function has been significantly improved in OpenVMS Version 7.1 for certain SCSI tape drives. The new IO$_SKIPFILE implementation functions correctly with all built-in OpenVMS tape functions such as INIT, MOUNT, BACKUP, and COPY when tapes are formatted according to the ANSI Standard X3.27-1987. This is the default tape standard for OpenVMS.

Higher performance for the IO$_SKIPFILE function is requested with a new modifier (IO$M_ALLOWFAST). When the IO$M_ALLOWFAST modifier is used, IO$_SKIPFILE stops at the end of data, rather than at double filemarks. For more information about the IO$M_ALLOWFAST modifier, refer to *OpenVMS Version 7.1 New Features Manual* and *OpenVMS I/O User's Reference Manual.*

The MTAACP and Backup utilities are compatible with the new behavior and have been modified to use the new IO$M_ALLOWFAST modifier. The behavior of MTAACP and Backup are unchanged when ANSI-formated tapes are used. They may behave differently when non-ANSI standard tapes are used. Other tape applications should be examined to determine whether they are compatible with the new semantics. If an application is compatible, or if it can be made compatible, the application should be modified to use the IO$M_ALLOWFAST modifier with the IO$_SKIPFILE function. If you have an application that is compatible with the new semantics, but it cannot be modified to use the IO$M_ ALLOWFAST modifier, refer to the documentation in SYS$ETC:MKSET.TXT.

6.3 ISA_CONFIG.DAT Unsupported in Future Release (Alpha Only)

V7.1

Support for using the SYS$MANAGER:ISA_CONFIG.DAT file to configure ISA devices will be discontinued in a future release of OpenVMS Alpha. If you use this file, you should convert to using the ISACFG utility from the console, and the new file-based autoconfiguration method for loading device drivers (as described in the *OpenVMS Version 7.1 New Features Manual*).

6.4 Required Change in ISA_CONFIG.DAT on AlphaStation 200/400

V7.1

Customers configuring ISA devices on AlphaStation 200/400 Family systems must change their SYS$MANAGER:ISA_CONFIG.DAT file, so that the node information for each device appears at the end of each device description block.

Warning

For upgrades from OpenVMS Version 6.2 or 7.0 systems, this change must be made before starting the upgrade procedure.

For example, prior to OpenVMS Version 7.1, the following device description block:

```
[AUA0]
 NAME=AU
 NODE=3
 DRIVER=SYS$MSBDRIVER
 IRQ=9
 DMA=(0,1)
 PORT=(388:4,530:8)
```

would be changed as follows:

```
[AUA0]
 NAME=AU
 DRIVER=SYS$MSBDRIVER
 IRQ=9
 DMA=(0,1)
 PORT=(388:4,530:8)
 NODE=3
```

Customers using SYS$MANAGER:ISA_CONFIG.DAT files should read Section 6.3.

6.5 Naming Serial Line Devices on Alpha Systems

V7.1

OpenVMS has changed the way it handles the second serial port on the following Alpha systems:

- AlphaStation 200, 400, 500 and 600 families
- AlphaServer 1000, 2000, 2100, and 4100 families

If one of these systems is booted with the *console* environment variable set to graphics, the name of the serial line (COM1) will be different from that in previous releases. The COM1 device is called TTB0 instead of OPA1. In this case, the COM1 device is controlled by SYS$YSDRIVER instead of SYS$OPDRIVER.

If the console is set to serial, the COM1 device is called OPA0.

6.6 Errors After a Reset on Some Wide SCSI Adapters (Alpha)

V7.1

In OpenVMS Alpha Version 7.1, wide SCSI disk drives connected to QLogic adapters such as KFTIA fast wide differential ports, KZPDA, KZPSM, and the AlphaStation 600 and AlphaServer 4100 embedded SCSI ports may encounter the problem described in this section.

An error such as SS$_CTLERR occasionally occurs following a bus reset on the QLogic drives. This error occurs only if a SCSI INQUIRY command is sent to the device immediately after the reset.

These errors are not likely to occur under normal circumstances because OpenVMS rarely follows a reset immediately by an INQUIRY command. The errors are most likely to be seen when running SYSMAN IO AUTOCONFIGURE immediately after a reset, or when issuing certain HSZTERM commands after an HSZ reset. If a wide device on any of the above adapters appears to be generating multiple errors after such a reset, the condition may be resolved by using any SCSI command other than INQUIRY. For example, this could be done by mounting the device or by running SYS$ETC:SCSI_INFO against it.

6.7 Problem with Exabyte EXB-8200 Tape Drives (VAX Only)

V7.1

Exabyte EXB-8200 tape drives do not work with OpenVMS VAX Version 7.1. If you have an Exabyte EXB-8200 tape drive or another third-party SCSI-1 tape drive on your VAX system, see the *Cover Letter for OpenVMS Alpha and OpenVMS VAX Version 7.1* for more information.

6.8 Memory Holes on AlphaServer 4100 Systems

V7.1

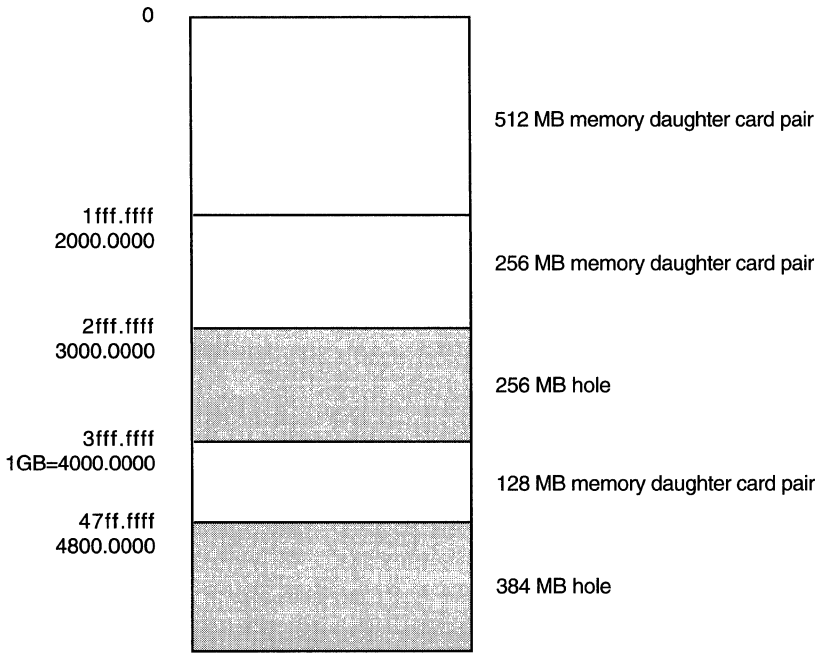
Physical memory holes might exist on AlphaServer 4100 systems. As illustrated in Figure 6–1, there are three different sizes of memory daughter card pairs: 512 MB, 256 MB, and 128 MB. In accordance with AlphaServer 4100 systems configuration rules, memory card pairs must be arranged in descending order of size.

The AlphaServer 4100 hardware reads the first set of memory daughter cards and assumes that any memory card pairs that follow are the same size. Memory holes occur because memory card pairs following the first set of cards read by the hardware might not be the same size. As shown in Figure 6–1, the hole at 3000.0000 must be dealt with by OpenVMS. The hole at 4800.0000 is at the top of the address space and can be ignored by OpenVMS.

Note

Previous versions of OpenVMS Alpha did not efficiently support systems with physical memory holes and ultimately led to an inefficient use of system memory. The memory management data structures in OpenVMS Alpha Version 7.1 have been slightly modified to recognize the memory holes. As a result, inefficiencies in previous versions of the OpenVMS Alpha operating system have been eliminated.

Figure 6–1 Example Memory Diagram



Note that this configuration impacts the algorithm used to determine whether a driver needs to use map registers. In releases prior to OpenVMS Alpha Version 7.1, device drivers do the following:

1. Call IOC$NODE_DATA with key IOC$K_DIRECT_DMA_SIZE to obtain the size of the direct DMA window (in megabytes). This is usually 1 GB.
2. Convert the size returned from IOC$NODE_DATA to pages and compare the size with **mmg$gl_memsize**, which contains the number of pages in physical memory.
3. If **mmg$gl_memsize** is greater than the size returned from IOC$NODE_DATA, use map registers; otherwise, use the direct DMA window.

The **mmg$gl_memsize** global cell does not contain the memory hole. As a result, the system has only 7/8 GB of memory, but according to the algorithm in releases prior to OpenVMS Alpha Version 7.1, it appears that the device can use the direct DMA window. Yet there is 128 MB of memory beyond the 1 GB border, which requires that the drivers use map registers. To eliminate this problem, drivers using the algorithm in releases prior to OpenVMS Alpha Version 7.1 must substitute it with the following algorithm:

1. Call IOC$NODE_DATA with key IOC$K_DIRECT_DMA_SIZE to obtain the size of the direct DMA window (in megabytes). This is usually 1 GB.
2. Convert the size returned from IOC$NODE_DATA to pages by dividing the number of bytes by the contents of **mmg$gl_page_size**. For example:

```
int dma_size;
int pages;

status = IOC$NODE_DATA (crb, IOC$K_DIRECT_DMA_SIZE, &dma_size);
        /* dma_size contains the number of megabytes.
         * convert number of megabytes to bytes.
         */
dma_size = dma_size * (1024 * 1024);
        /* Convert number of bytes to number of pages by
         *  dividing by number of bytes per page.
         */
pages = dma_size / MMG$GL_PAGE_SIZE;
```

3. Compare the resulting number of pages with **mmg$gl_maxpfn + 1**.
4. If **mmg$gl_maxpfn + 1** is greater than the size returned from IOC$NODE_DATA, use map registers; otherwise use the direct DMA window.

6.9 SYS$MSBDRIVER Removed from OpenVMS Alpha Distribution

V7.0

The driver for the Microsoft Windows Sound System ISA sound card (MSB), SYS$MSBDRIVER, has been removed from the OpenVMS Alpha distribution as of Version 7.0. The following files have been removed:

- SYS$LOADABLE_IMAGES:SYS$MSBDRIVER.EXE
- SYS$EXAMPLES:SOUND_SERVICES.C
- SYS$EXAMPLES:SOUND_SAMPLE.C
- SYS$EXAMPLES:SOUND_SAMPLE.SND
- SYS$LIBRARY:SYS$STARLET_C.TLB module MSB.H

An enhanced version of this driver, called MMOV$MSBDRIVER, is included in Multimedia Services Version 2.0 for OpenVMS Alpha. This layered product also includes support for video capture and playback, an enhanced version of DECsound, and other audio and video applications.

MMOV$MSBDRIVER provides the same $QIO programming interface as SYS$MSBDRIVER. Digital recommends that the WAVE Applications Programming Interface provided by Multimedia Services for OpenVMS be used instead because it is more flexible and is portable to other platforms. (Multimedia Services Version 2.0 for OpenVMS is described in SPD 64.24.00.)

6.10 Device IPL Setup for OpenVMS Alpha Drivers

V6.2

Alpha hardware platforms that support PCI, EISA, and ISA buses deliver I/O device interrupts at different IPLs, either 20 or 21. The IPL at which device interrupts are delivered can change if you move the device from one platform to another. This is a problem if the driver declares its device IPL to be 20, and then that driver is executed on a machine that delivers I/O device interrupts at IPL 21.

The simplest solution to this problem is for PCI, EISA, and ISA device drivers to use IPL 21. This works correctly on platforms that deliver I/O device interrupts at IPL 20 and on platforms that deliver I/O device interrupts at IPL 21.

A future release of OpenVMS Alpha may provide a platform-independent mechanism for drivers to determine the device IPL dynamically.

6.11 AlphaStation 255: PCI Configuration Restriction

V7.1

The OpenVMS Alpha operating system does not support PCI option cards configured in PCI slot 0 on any AlphaStation 255 series systems.

PCI slot 0 is the lowest physical PCI option slot on AlphaStation 255 series systems. The interrupt signal for this slot is shared with the built-in Ethernet port. Because the OpenVMS Alpha operating system does not currently permit PCI devices to share an interrupt line, a PCI device installed in slot 0 will not function correctly or might cause errors to occur with the built-in Ethernet port. As a result of this restriction, AlphaStation 255 series systems support a maximum of two PCI option cards, configured in slot 1 and slot 2.

6.12 Recommendation for RZ25M and RZ26N Disk Drives (Alpha)

V7.1

During the testing of Digital supported SCSI disk drives on configurations with DWZZAs and long differential SCSI buses, two drives (RZ25M and RZ26N) were found to have bus phase problems. For this reason, these drives are not recommended for use in configurations where the differential bus length connecting DWZZAs equals or exceeds 20 meters.

This recommendation applies only to the RZ25M and RZ26N drives. All other disk drives, listed as supported in the OpenVMS SPD, may be used in configurations to the full bus lengths of the SCSI-2 specification.

6.13 SCSI Controller Restriction on AlphaServer 2100 Systems

V6.2

The Adaptec 1740/1742 SCSI controller (PB2HA–SA) is not supported on AlphaServer 2100 systems having more than 1 gigabyte (GB) of memory. If the controller is connected to such a system, the following message appears on the operator's console:

```
%PKJDRVR-E- PKX0, Port is going OFFLINE.
```

6.14 OpenVMS Alpha SCSI Firmware Support

The following sections relate to SCSI firmware support.

6.14.1 Recommended Firmware Support for RZ26N and RZ28M Disks

V6.2—1H3

The minimum firmware revision level recommended for RZ26N and RZ28M disks is Revision 0568.

If the latest firmware revision level is not used with these disks, multiple problems may occur.

6.14.2 Required Firmware for Multi-Host Use of RZ26L and RZ28 Disks

V6.2

If you install RZ26L or RZ28 disks on a multi-host SCSI bus in an OpenVMS Cluster, the disk's minimum firmware revision is 442.

The following sections describe a procedure that can be used to update the firmware on some RZ26L and RZ28 drives. This procedure can only be used with drives that are directly connected to a SCSI adapter on a host system. Drives that are attached through an intelligent controller (such as an HSZ40 or KZPSC) cannot be updated using this procedure. Refer to the intelligent controller's documentation to determine whether an alternative firmware update procedure exists.

Important Note

Only certain RZ26L and RZ28 firmware revisions can be safely upgraded to firmware revision level 442. Refer to Section 6.14.2.1 to determine if your disks are capable of being upgraded to firmware revision level 442. If your disk is capable of supporting firmware revision level 442, use the RZTOOLS Utility that is described in Section 6.14.2.2 to update the disk's firmware.

6.14.2.1 Firmware Revision Level 442 Requirements

Only the combinations of disk drives and firmware revision levels listed in Table 6–1 are capable of being upgraded safely to firmware revision level 442. Performing the update procedure on any other combination can permanently damage the disk.

Table 6–1 Revision Level 442 Firmware Compatibility

Disk Drive	Firmware Revision	Disk File Name
RZ26L	440C	RZ26L_442D_DEC.FUP
RZ28	441C or D41C	RZ28_442D_DEC2104.FUP
	435 or 436	RZ28P4_442C_DEC.FUP

6.14.2.2 Firmware Revision Level 442 Installation Procedure

If you determine that your disk requires revision level 442 firmware, and it is capable of being upgraded safely, use the following procedure to update the firmware. (See Table 6–1 for the file name of the disk you are upgrading.)

```
$ MCR SYS$ETC:RZTOOLS_ALPHA DKB500 /LOAD=SYS$ETC:filename.FUP
  Read in 262144 bytes.
  Current FW version - X440C
  Upgrading to       - DEC0
  Loading code  ......
  New code has been sent to the drive.
```

6.15 OpenVMS Alpha SCSI Port and Class Drivers

V6.2

The following sections describe OpenVMS Alpha SCSI class and port device driver restrictions.

6.15.1 Add-On SCSI Adapters

V6.2

Version 6.2 and later of OpenVMS Alpha supports various add-on SCSI adapters. Digital's AlphaGeneration platforms typically support one or more integral SCSI adapters, with the option of installing additional add-on SCSI adapters. Due to differences in device-naming conventions used between the Alpha console and OpenVMS, the OpenVMS device name may not match the name displayed by the console.

For example, the console designation for a SCSI device on the integral SCSI adapter may be DKA100. However, when two additional add-on SCSI adapters are added, the "A" designation becomes "C"; and DKA100 appears as DKC100 when OpenVMS is running.

Note that although the console and OpenVMS device names may be different, the unique specification of a device name from the console to the device name under OpenVMS will stay consistent, provided add-on SCSI adapters are not added or removed.

For information about device naming on certain OpenVMS Alpha systems, see Appendix A.

6.16 OpenVMS VAX Device Support Documentation Corrections

This section describes corrections to OpenVMS VAX device support documentation.

6.16.1 *OpenVMS VAX Device Support Manual*

V6.1

The following sections describe corrections to the *OpenVMS VAX Device Support Manual*. (This manual has been archived but is available in PostScript and DECW$BOOK (Bookreader) formats on the OpenVMS Documentation CD–ROM. A printed book can be ordered through DECdirect (800–344–4825).)

6.16.1.1 Linking a Device Driver

Chapter 12 of the *OpenVMS VAX Device Support Manual*, Version 6.0, describes how to assemble, link, and load a device driver. In step 3 of the procedure for preparing a driver for loading into the operating system, append the following text to the end of the procedure (following the paragraph that begins: "The resulting image must . . . "):

To produce an image with a symbol table compatible with the System Dump Analyzer (SDA), you must link again; this time, using the UNIVERSAL=* option statement (to include all global symbols and to ensure proper state of the REL bits in the object records). Relink as shown in the following example:

```
$ LINK /NOEXECUTABLE/NOTRACEBACK/NOSYSSHR -
_$     /SYMBOLS=MYDRIVER.EXE,-
_$     /SHARE=DUMMY_FILE_NAME,-
_$     /NOMAP,MYDRIVER1.OBJ,MYDRIVER2.OBJ,-
_$     SYS.STB/SELECTIVE,-
_$     SYS$INPUT/OPTION
_$     BASE=0
_$     UNIVERSAL=*
```

For more information about the Linker, see the *OpenVMS Linker Utility Manual*.

6.16.1.2 Device-Register I/O Space: Usage Restrictions

Chapter 5 of the *OpenVMS VAX Device Support Manual*, Version 6.0, describes device driver coding and the restrictions on the use of device-register I/O space. The third sentence of the fifth bulleted paragraph in Section 5.2 states that the instructions that refer to UNIBUS adapter registers must use longword context. This is the wrong bus. The sentence should read:

> "Instructions that refer to MASSBUS adapter registers must use the longword context."

6.16.2 *OpenVMS VAX Device Support Reference Manual*

V6.1

The following sections describe corrections to the *OpenVMS VAX Device Support Reference Manual*. (This manual has been archived but is available in PostScript and DECW$BOOK (Bookreader) formats on the OpenVMS Documentation CD–ROM. A printed book can be ordered through DECdirect (800–344–4825).)

6.16.2.1 COM$DRVDEALMEM Routine Synchronization

Chapter 3 of the *OpenVMS VAX Device Support Reference Manual*, Version 6.0, contains a section describing the COM$DRVDEALMEM routine.

At the end of the paragraph under *Synchronization*, add the following sentence:

"If called at IPL$_SYNCH or higher, the routine executes the fork process."

6.16.2.2 CRB Data Structure

Chapter 1, Section 1.7, of the *OpenVMS VAX Device Support Reference Manual*, Version 6.0, contains Table 1-8 describing the CRB data structure fields. The description in the table for the CRB$L_INTD field is confusing and needs clarification. Replace the first two sentences in the CRB$L_INTD description as follows:

Field Name	Description
CRB$L_INTD	Portion of the interrupt transfer vector block that stores executable code, driver entry points, and I/O adapter information. This 10-longword area is overlaid with the contents of the interrupt transfer vector block that starts at VEC$L_INTD (offset 16) as described in Section 1.7.1. It contains pointers to the driver's . . .

6.16.2.3 SCDRP Data Structure SCSI Flags

Chapter 1 of the *OpenVMS VAX Device Support Reference Manual*, Version 6.0, contains a section describing the SCDRP data structure SCSI flags.

In the SCDRP$L_SCSI_FLAGS field description for bit SCDRP$V_LOCK, make the following correction:

Change: SCDRP$VLOCK

To: SCDRP$V_LOCK

6.16.2.4 SPI$CONNECT Macro

Chapter 2 of the *OpenVMS VAX Device Support Reference Manual*, Version 6.0, contains a section describing the SPI$CONNECT macro.

In the table listing the required inputs, add:

R4 Address of the SPDT

In the values returned in R3, the SPDT$M_CMDQ bit was added to the port capability mask (SPDT$L_PORT_FLAGS). When set, SPDT$M_CMDQ indicates that the port driver supports command queuing I/O.

In the return values table listing R3 and the mask bits (after SPDT$M_LUNS), add:

SPDT$M_CMDQ Supports command queuing I/O

6.16.2.5 SPI$GET_CONNECTION_CHAR and SPI$SET_CONNECTION_CHAR Macros

Chapter 2 of the *OpenVMS VAX Device Support Reference Manual*, Version 6.0, contains sections describing the SPI$GET_CONNECTION_CHAR and SPI$SET_CONNECTION_CHAR macros. Appended to the macro characteristics buffer is longword #12 for SCSI-2 support.

At the end of the characteristics buffer table in these macro descriptions, add the longword #12 information as follows:

12 SCSI-2 device characteristic status bits. Bits of this longword are defined as follows:

- When Bit 0 is set, (SCDT$V_SCSI_2) indicates the device connection is SCSI-2 conformant.
- When Bit 1 is set, (SCDT$V_CMDQ) indicates the device connection supports command queuing.

6.16.2.6 $EQULST Macro

Chapter 2 of the *OpenVMS VAX Device Support Reference Manual*, Version 6.0, contains a section describing the $EQULST macro.

In the parameter description for **symbol,value** insert the phrase *in decimal* as follows:

" . . . and value specifies in decimal the value of the symbol."

A

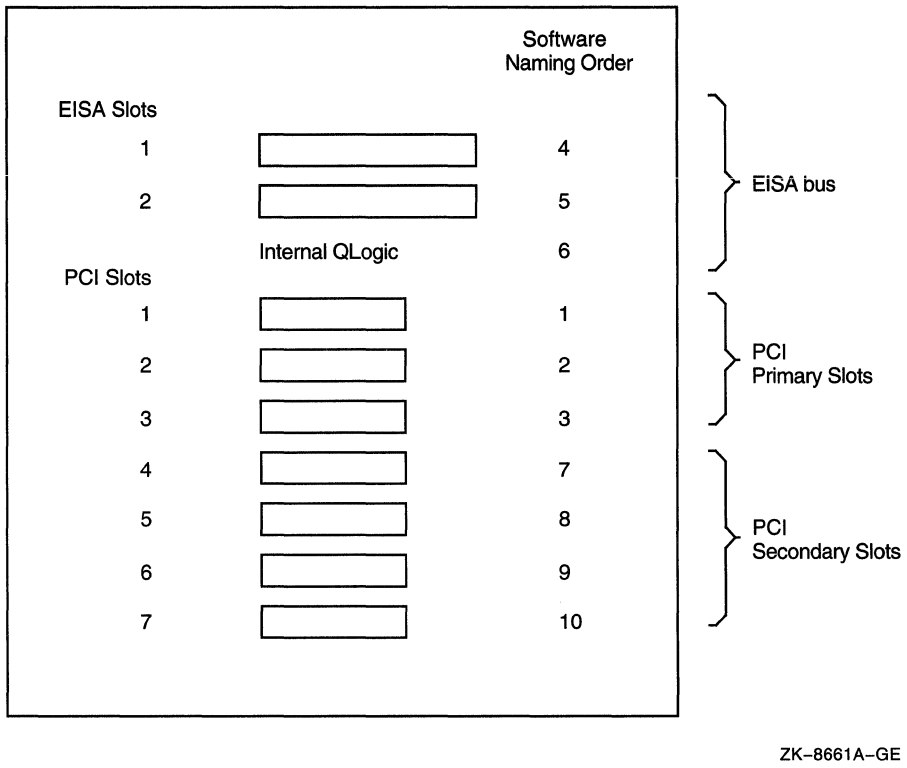
OpenVMS Alpha Version 6.2—1H3 Device Naming Notes

A.1 Device Naming on AlphaServer 1000A Systems

AlphaServer 1000A systems have seven PCI option slots and two EISA option slots, in addition to an integrated QLogic 1020A Fast Wide SCSI controller. The OpenVMS operating system and the console do not always assign the same names to devices in these option slots. In addition, the slot numbering convention used by the console and the OpenVMS operating system does not match the physical slot numbers on the backplane.

The EISA slots are connected using a PCI-to-EISA bridge on the primary PCI bus. The upper three PCI option slots (numbered 1 to 3) are also on the primary PCI bus, while the lower four PCI option slots (numbered 4 to 7) are connected using an integrated PCI-to-PCI bridge. The QLogic 1020A SCSI controller is in a fixed position relative to the option slots; this controller is always configured as the first device behind the integrated PCI-to-PCI bridge, preceding slots 4 to 7. (See Figure A–1.)

Figure A–1 AlphaServer 1000A Hardware Backplane



The OpenVMS operating system assigns device names in the following order:

1. Devices on the primary PCI (slots 1 to 3)
2. EISA devices
3. Integrated SCSI controller
4. Devices on the secondary PCI (slots 4 to 7)
5. Devices on PCI-to-PCI bridges in secondary PCI slots
6. Devices on PCI-to-PCI bridges in primary PCI slots

Consider a system with only a KZPSA SCSI controller in slot 1, the first slot on the primary PCI bus. The KZPSA appears as PKA, and the integrated QLogic 1020A appears as PKB. If another KZPSA were added in slot 4, behind the integrated bridge, it would appear as PKC. If yet another KZPSA were added in slot 3, the third slot on the primary PCI bus, the device names would shift; the original KZPSA in slot 1 remains PKA, the new KZPSA in slot 3 becomes PKB, the QLogic becomes PKC, and the KZPSA in slot 4 becomes PKD.

In another example, consider a system with one KZPSA in slot 4. This KZPSA is behind the integrated bridge, behind the QLogic 1020A, and appears as PKB with the QLogic 1020A as PKA. A system with only bridge cards in its option slots, for instance, P1SE bridges with integrated SCSI controllers, would always see the QLogic 1020A as PKA and the SCSI controllers on the P1SE bridges as PKB and so forth.

A.1.1 Restriction for PCI Adapters

As shown in Figure A–1, PCI slots 4-7 on AlphaServer 1000A systems are connected by an integrated PCI-to-PCI bridge. Certain PCI adapters, such as the KZPSM, also contain an onboard PCI-to-PCI bridge. Because the KZPSM is not supported in PCI slots 4-7, you must place that adapter in PCI slots 1, 2, or 3.

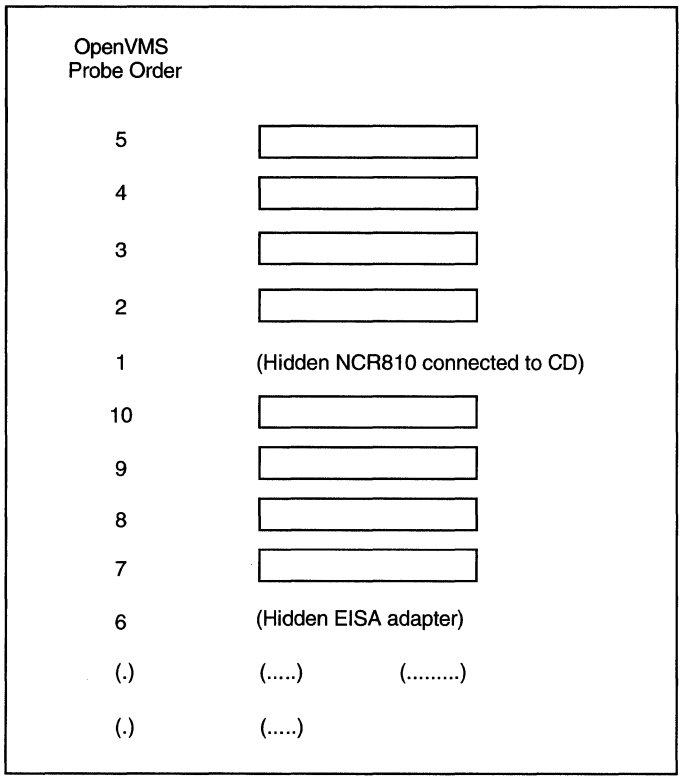
A.2 Device Naming on AlphaServer 4100 Systems

AlphaServer 4100 systems have eight PCI option slots, three of which are shared with EISA option slots. Because the OpenVMS operating system and the console each have their own device naming conventions, the console name of a device will not necessarily match the OpenVMS name of a device. In addition, the slot numbering convention used by the console and the OpenVMS operating system does not match the physical slot order on the backpanel.

AlphaServer 4100 systems also include an integrated NCR810 SCSI controller and an integrated PCI/EISA bridge. The integrated NCR810 SCSI controller is connected directly to the compact disc reader.

The order in which OpenVMS names devices is shown by the OpenVMS Probe Order column in Figure A–2.

Figure A–2 AlphaServer 4100 Hardware Backpanel



ZK–8684A–GE

The integrated NCR810 controller (connected to the compact disc reader) will always be assigned the same name by both the console and by OpenVMS because it is always named first. The OpenVMS operating system assigns device names in the following order (using the OpenVMS probe numbers shown in Figure A–2):

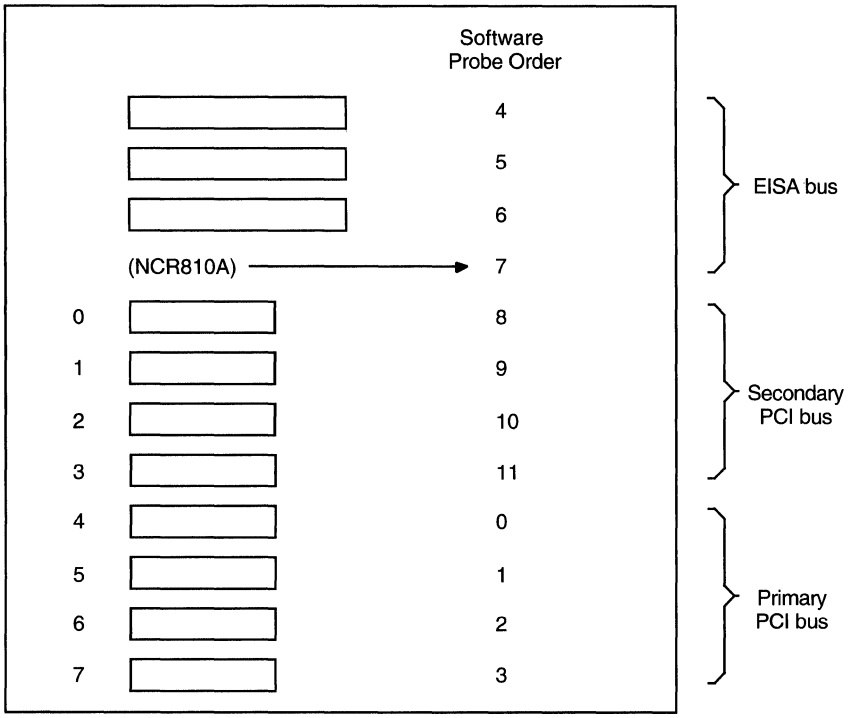
1. Integrated NCR 810 connected to the CD
2. PCI slots 2 through 5
3. Devices on PCI-to-PCI bridges on PCI slots 2 through 5
4. PCI slots 7 through 10
5. EISA slots 7 through 9
6. Devices on PCI-to-PCI bridges on PCI slots 7 through 10

A.3 Device Naming on AlphaServer 2100A Systems

AlphaServer 2100A systems have eight PCI option slots and three EISA option slots, in addition to an integrated NCR810A SCSI controller. The OpenVMS operating system and the console do not always assign the same names to devices in these option slots. In addition, the slot numbering convention used by the console and by the OpenVMS operating system does not match the physical slot numbers on the backplane.

The EISA slots are connected using a PCI-to-EISA bridge on the primary PCI bus. The bottom four PCI option slots (hardware numbers 4 to 7) are also on the primary PCI bus, while the remaining four PCI option slots (the top slots, hardware numbers 0 to 3) are connected using an integrated PCI-to-PCI bridge. The NCR810A SCSI controller is in a fixed position relative to the option slots; the NCR810A controller is always configured as the first device behind the integrated PCI-to-PCI bridge, preceding PCI slots 0 to 3. (See Figure A–3.)

Figure A–3 AlphaServer 2100A Hardware Backplane



ZK–8563A–GE

The OpenVMS operating system assigns device names in the following order:

1. Devices on the primary PCI (slots 4 to 7)
2. EISA devices
3. Integrated NCR810A SCSI controller
4. Devices on the secondary PCI (slots 0 to 3)
5. Devices on PCI-to-PCI bridges in secondary PCI slots
6. Devices on PCI-to-PCI bridges in primary PCI slots

Consider a system with only a KZPSA SCSI controller in slot 4, the first slot on the primary PCI bus. The KZPSA appears as PKA, and the integrated NCR810A appears as PKB. If another KZPSA were added in slot 0, behind the integrated bridge, it would appear as PKC. If yet another KZPSA were added in slot 6, the third slot on the primary PCI bus, the device names would shift; the original KZPSA in slot 4 remains PKA, the new KZPSA in slot 6 becomes PKB, the 810A becomes PKC, and the KZPSA in slot 0 becomes PKD.

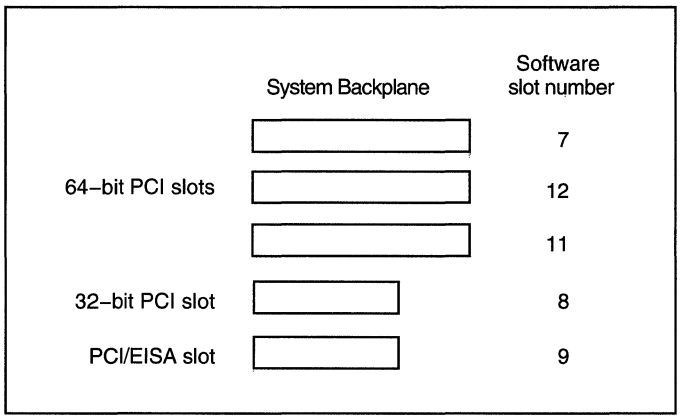
In another example, consider a system with one KZPSA in slot 0. This KZPSA is behind the integrated bridge, behind the 810A, and appears as PKB, with the 810A as PKA. A system with only bridge cards in its option slots, for instance, P1SE bridges with integrated SCSI controllers, would always see the 810A as PKA and the SCSI controllers on the P1SEs as PKB and so forth.

A.4 Device Naming on AlphaStation 600 Systems

AlphaStation 600 Series systems have three 64-bit PCI option slots, one 32-bit PCI option slot, and one option slot that can be used either as a 32-bit PCI option slot or as an EISA slot. (See Figure A–4.)

The OpenVMS operating system and the console do not always assign the same names to devices in these option slots.

Figure A–4 AlphaStation 600 Hardware Backplane



ZK–8957A–GE

The OpenVMS operating system assigns device names in the following order:

1. Devices in slots 7, 8, 9, 11, and 12
2. Devices on PCI-to-PCI bridges in slots 7, 8, and 9
3. EISA devices
4. Devices on PCI-to-PCI bridges in slots 11 and 12

Consider a system with a KZPSA SCSI controller in slot 8 and a P2SE PCI-to-PCI bridge card with two SCSI controllers in slot 9. The KZPSA will appear as PKA, and the P2SE SCSI controllers will appear as PKB and PKC. Swapping the cards so that the KZPSA is in slot 9 and the P2SE is in slot 8 will not change the device names. The KZPSA will still appear as PKA, and the P2SE SCSI controllers will still appear as PKB and PKC.

B

Remedial Kits Included in OpenVMS Version 7.1

This appendix lists remedial kits that are included in OpenVMS Version 7.1.

Digital updates existing kits and creates new kits as necessary. Contact your Digital support representative for the latest information about new remedial kits.

The following sections list the remedial kits included in Version 7.1 of the OpenVMS VAX and OpenVMS Alpha operating systems. If you used to install one of the kits listed here, you will not need to do so with Version 7.1.

Kit names are constructed from the following information in this order:

- Platform name: VAX or ALP (for Alpha)
- Facility name, abbreviated to 4 characters if necessary
- Number of the kit for this facility for this version
- Version number

For example, VAXUAF01_070 is the first remedial kit created to correct the Authorize utility that shipped in Version 7.0 of OpenVMS VAX.

B.1 Remedial Kits Included in OpenVMS Alpha Version 7.1

The following remedial kits are included in Version 7.1 of the OpenVMS Alpha operating system:

ALPACRT04_070
ALPACRT08_061
ALPAMAC02_062
ALPAMAC02_070
ALPAMAC03_061
ALPBOOT03_062
ALPBOOT05_062
ALPBOOT06_061
ALPBOOT06_062
ALPCLIU01_070
ALPCLUS01_070
ALPCMAR04_062
ALPCPU02_062
ALPCPU905_062
ALPCPUC04_062
ALPDDTM03_070
ALPDEBU05_070
ALPDRIV04_070
ALPDRIV05_062
ALPDRIV06_062

ALPDRIV13_061
ALPDWMW01_U3012
ALPDWXT01_070
ALPF11C03_062
ALPF11X03_070
ALPIMGD02_061
ALPINIT01_070
ALPLAD03_070
ALPLAN01_070
ALPLAN03_062
ALPLAN05_061
ALPLAT01_062
ALPLOAD01_070
ALPLOAD02_070
ALPLOGI02_070
ALPMAIL01_070
ALPMANA01_070
ALPMOTF01_U4012
ALPMOTF07_U3012
ALPMOUN01_062
ALPMSCP01_070
ALPMTAA01_070
ALPOPCO01_070
ALPOPDR02_070
ALPPRTS01_070
ALPPRTS02_070
ALPPTD01_070
ALPQMAN02_070
ALPRAMP01_061
ALPRMS01_062
ALPRMS01_070
ALPRMS03_061
ALPSCSI02_070
ALPSCSI03_062
ALPSCSI04_061
ALPSHAD05_062
ALPSHAD12_061
ALPSYS02_070
ALPSYS03_070
ALPSYS04_070
ALPSYS06_062
ALPSYS06_070
ALPSYS07_070
ALPSYS08_070
ALPSYS18_061
ALPSYSL01_070
ALPTTDR01_070
ALPTTDR02_070

B.2 Remedial Kits Included in OpenVMS VAX Version 7.1

The following remedial kits are included in Version 7.1 of the OpenVMS VAX operating system:

VAXBADB02_070
VAXCDU01_061
VAXCLIU01_070
VAXCLIU02_070
VAXCLIU03_062
VAXDDTM01_070
VAXDEBU01_070
VAXDEBU04_070
VAXDEBU05_070
VAXF11X03_070
VAXINIT01_070
VAXLMF01_U2055
VAXLOAD02_070
VAXLOGI02_070
VAXMAIL01_070
VAXMANA02_070
VAXMOTF01_U4012
VAXMSCP01_070
VAXOPCO01_070
VAXPHV01_062
VAXPHV01_070
VAXPHV02_062
VAXPHV02_070
VAXPHV03_061
VAXPHV04_061
VAXPRTS02_070
VAXPTD01_070
VAXQMAN02_070
VAXRMS01_062
VAXRMS02_062
VAXRMS03_061
VAXSMGR02_070
VAXSYS03_070
VAXSYS04_070
VAXSYS05_062
VAXSYS05_070
VAXSYS06_070
VAXSYS07_062
VAXSYSL01_070
VAXTTDR01_070
VAXUAF01_070

C

Product Retirement Notices

This appendix contains notifications about OpenVMS products that are no longer supported or that are slated for retirement. It also lists manuals that have been archived with this release.

C.1 DEC 3000 Series Workstations: PXG Graphics Board Not Supported (Alpha Only)

V7.0

Starting with OpenVMS Alpha Version 7.0, the PXG graphics board is no longer supported.

C.2 DECthreads: POSIX 1003.4a Draft 4 Interface To Be Retired

V7.0

The POSIX 1003.4a, Draft 4 (or "d4") interface of DECthreads is slated for retirement in a future release. Applications that were written using the POSIX 1003.4a, Draft 4 interface should be migrated to the new POSIX 1003.1c standard (or "pthread") interface provided by DECthreads. A compatibility mode for the Draft 4 POSIX 1003.4a interface has been provided in this release to help ease migration. This compatibility mode will be removed in a future release.

C.3 DTK$ (DECtalk) Available as Freeware

V7.1

With OpenVMS Version 7.1, the DECtalk facility (DTK$) is moved from maintenance status to retired status. The source code for this library will be moved to freeware status and will be available with the release of Version 7.1 from the following sources for those interested in doing their own development and support:

- On the freeware CD–ROM that ships with the OpenVMS operating system, starting with Version 7.1.
- On the World Wide Web at the following URL:

  `http://www.openvms.digital.com/openvms/freeware/index.html`

Starting with OpenVMS Version 7.1, Digital will no longer accept or act on CLDs posted against the DTK$ library.

C.4 Error Reporting Formatter (ERF) To Be Retired

V7.1

The Error Reporting Formatter (ERF) still ships with the current release of the operating system, but it will be retired in a future OpenVMS release. DECevent now replaces ERF as the bit-to-text translating tool for fault management on both VAX and Alpha systems.

C.5 ISA_CONFIG.DAT Unsupported in Future Release (Alpha Only)

V7.1

Support for using the SYS$MANAGER:ISA_CONFIG.DAT file to configure ISA devices will be discontinued in a future release of OpenVMS Alpha. If you use this file, you should convert to using the ISACFG utility from the console, and the new file-based autoconfiguration method for loading device drivers (as described in the *OpenVMS Version 7.1 New Features Manual*).

C.6 POLYCENTER Software Installation Utility: DECwindows Motif Interface To Be Retired

V7.1

The DECwindows Motif interface for the POLYCENTER Software Installation Utility will be retired in the next OpenVMS release. All functions of the POLYCENTER Software Installation utility will still be available from the DCL interface using the PRODUCT command. Starting with OpenVMS Version 7.1, the method of accessing the Motif interface has been changed to require using the new /INTERFACE=DECWINDOWS qualifier. For details, see Section 4.16.1.1.1.

C.7 PPL$ (Parallel Processing Library) Available as Freeware

V7.1

With OpenVMS Version 7.1, the Parallel Processing Library (PPL$) moves from maintenance mode to unsupported status. The source code for this library will be moved to freeware status and will be available with the release of Version 7.1 from the following sources for those interested in doing their own development and support:

- On the freeware CD–ROM that ships with the OpenVMS operating system, starting with Version 7.1.
- On the World Wide Web at the following URL:

  `http://www.openvms.digital.com/openvms/freeware/index.html`

Starting with OpenVMS Version 7.1, Digital will no longer accept or act on CLDs posted against the PPL$ library.

If you have used PPL$, consider whether you can use the OpenVMS DECthreads library to implement concurrent applications. The DECthreads library has shipped with the OpenVMS VAX operating system since Version 5.5 and with the OpenVMS Alpha operating system since Version 1.0. Starting with OpenVMS Version 7.0, DECthreads has supported multiprocessing as part of its multithreading capabilities. In addition, DECthreads implements the industry-standard POSIX multithreading interface. For more information about DECthreads, see the *Guide to DECthreads*.

C.8 Selected $QIO Function Codes To Be Retired

V7.1

The $MOUNT and $DISMOU system services are the preferred method for mounting and dismounting shadow sets. In the past, Digital has also supported the use of $QIO function codes IO$_CRESHAD, IO$_ADDSHAD, and IO$_ REMSHAD for these functions, although they were seldom, if ever, used. After this release, Digital will no longer support these function codes.

If any of your programs use these function codes, Digital recommends that you replace them with $MOUNT or $DISMOU system service calls, as appropriate.

C.9 Snapshot Facility Removed (VAX Only)

V7.1

With the release of OpenVMS VAX Version 7.1, the Snapshot facility has been removed. Snapshot was introduced with OpenVMS VAX Version 6.0 and was supported only on nonclustered VAX workstations. Enhancements to the OpenVMS operating system have made it impractical to continue maintaining this little used facility in OpenVMS VAX. (Snapshot was never available on Alpha systems.)

C.10 Archived Manuals

V7.1

Archived manuals are no longer maintained and are not part of the OpenVMS Documentation Set. However, they are available in PostScript and DECW$BOOK (Bookreader) formats on the OpenVMS Documentation CD–ROM. You can also order printed manuals through DECdirect (800–354–4825).

Table C–1 lists the manuals that have been archived starting with OpenVMS Version 7.1.

Table C–1 Manuals Archived With OpenVMS Version 7.1

Manual Title	File Name on CD–ROM
OpenVMS Glossary	OVMS_GLOSSARY.PS
OpenVMS Software Overview	OVMS_SW_OVERVIEW.PS
OpenVMS Programming Environment Manual	OVMS_PROG_ENVIRON.PS
Building Dependable Systems: The OpenVMS Approach	BUILD_DEPEND_SYS.PS
OpenVMS Compatibility Between VAX and Alpha	OVMS_COMPAT_VAX_ALPHA.PS
A Comparison of System Management on OpenVMS AXP and OpenVMS VAX	COMP_SYSMAN_AXP_VAX.PS
Migrating to an OpenVMS AXP System: Planning for Migration	MIG_AXP_PLAN_MIG.PS
Migrating an Environment from OpenVMS VAX to OpenVMS Alpha	OVMS_MIG_ENVIRON.PS
Migrating to an OpenVMS AXP System: Recompiling and Relinking Applications	MIG_AXP_RECOMP_RELINK.PS
Guide to OpenVMS Performance Management	GD_VAX_PERF_MAN.PS
Guide to OpenVMS AXP Performance Management	GD_AXP_PERF_MAN.PS
OpenVMS VAX Device Support Manual	OVMS_VAX_SUP_GD.PS
OpenVMS VAX Device Support Reference Manual	OVMS_VAX_SUP_REF.PS
Creating an OpenVMS Alpha Device Driver from an OpenVMS VAX Device Driver	CR_AXP_STEP2_DR_FR_VAX_DR.PS
Creating an OpenVMS AXP Step 2 Device Driver from a Step 1 Device Driver	CR_AXP_STEP2_DR_FR_STEP1.PS

Index

A

B

C

D

E

F

G

H

I

K

L

M

N

O

T

AA-QSBTB-TE

Printed in U.S.A.

Printed on
Recycled paper