HP OpenVMS Alpha Version 7.3-2 New Features and Documentation Overview

OpenVMS Alpha 7.3-2

This is a new manual.

Manufacturing Part Number: AA-RV8XA-TE

September 2003

© Copyright 2003 Hewlett-Packard Development Company, L.P.

Legal Notice

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

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Adobe®, Acrobat®, and Postscript® are registered trademarks of Adobe Systems Incorporated.

Java™ is a U.S. trademark of Sun Microsystems, Inc.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of the Open Group.

The HP OpenVMS documentation set is available on CD-ROM.

ZK6667

1. Summary of HP OpenVMS Alpha 7.3-2 New Features

Performance Features . . . 17
Security Features . . . 17
UNIX Portability Features . . . 18
System Management Features . . . 18
Summary Tables . . . 19

2. General User Features

Blocks-to-Bytes Display Option . . . 25
COPY and SEARCH Performance Enhancements . . . 26
DCL Command Size Limits Increased . . . 27
Unchanged Size Limits . . . 27
DCL Commands and Lexical Functions . . . 27
Documentation Updates on the World Wide Web . . . 31
MIME Utility Enhancements . . . 31

3. System Management Features

AlphaServer Systems . . . 33
AlphaServer System DS15 Support . . . 34
AlphaServer Series Systems ES47, ES80, and GS1280 Support . . . 34
Backup Utility Changes and Additions . . . 34
New BACKUP Command Qualifiers . . . 34
New BACKUP Option Structure Types . . . 35
Change to BCK_OPT_K_UNSHELVE . . . 35
New BACKUP API Example . . . 36
Component Indictment . . . 37
Overview of Indictment Process . . . 38
Sample SYS$INDICTMENT_POLICY.COM . . . 38
Dissimilar Device Shadowing and Dynamic Volume Expansion . . . 39
Error Log Viewer (ELV) Utility . . . 40
Euro Symbol and Additional Font Sets Support . . . 40
Extended File Cache (XFC) Enhancements . . . 41
XFC Performance Enhancements . . . 41
XFC Usability Improvements . . . 42
Fast Path for LAN Devices . . . 42
Fast Path Support for Smart Array 53xx Series Adapter (KZPDC) . . . 43
Fibre Channel New Features . . . 43
MSA1000 Support . . . 43
New FIBRE_SCAN Utility . . . 43
Gigabit Ethernet Adapter (DEGXA) Support . . . 44
Simplified Installation, Upgrade, and Configuration Procedures . . . 44
CDSA and Kerberos in OpenVMS Upgrades and Installations . . . 44
Enhanced Time Zone Selection Procedure (UTC$TIME_SETUP.COM) . . . 44
LAN Failover . . . 45
failSAFE IP . . . 45
New Mail Flag . . . 46

Mount Verification Message Reduction 46
PEdriver Fast Path Support 47
POLYCENTER Software Installation New Features 47
Running in Recovery Mode 47
Uninstalling a Patch 47
Data Compression 48
Display of File and Module Generation Numbers 49
SET DEVICE Command Can Reset Device Counts 49
SYSMAN DUMP_PRIORITY Commands 49
System Parameters 50
BALSETCNT and WSMAX System Parameters 50
SHADOW_REC_DLY System Parameter 50
SHADOW_SITE_ID System Parameter 50
UNIX Portability Security Features 50
USB Configuration Manager (UCM) Utility 51
Volume Shadowing Management Functions 51

4. Programming Features

Authentication and Credentials Management Extensions (ACME) Agent Software Development Kit (SDK) 53
C Run-Time Library Enhancements 54
POSIX Style Identifiers 54
Socket Enhancements 54
Set Default Directory for Child Processes 54
New Logicals 55
Performance Enhancements 55
Additional Enhancements 56
New and Changed Functions 56
DDT Intercept Establisher Routines and Device Configuration Notification Routines 57
DDT Intercept Establisher Routines 57
Intercept Levels 58
Restrictions 58
Routines 59
Device Registration Callback Routines 62
Device Configuration Callback Routine 64
HP DECdtm Version 2.1 65
$GETDVI Item Code DVI$SHDW_SITE 66
$GETRMI System Service Calls 66
Large-Pool Lookaside Lists 66
POSIX Threads NUMA Support 66
New System Dump Analyzer (SDA) Commands 67
New and Revised System Services 69
New System Services 70
SYS$GOTO_UNWIND_64 (Alpha and I64 Only) 70
SYS$SET_RETURN_VALUE (Alpha and I64 Only) 70

5. Associated Products Features

ATI RADEON 7500 Graphics 71
Multiple RADEON Card Support 72
Supported Video Modes 72
Monitor Support 73
3D Direct Rendering 73
RADEON 7500 TCL Support 73
Common Data Security Architecture Version 2.0 Enhancements 73
OpenVMS e-Business Infrastructure Package 74
HP Kerberos Version 2.0 for OpenVMS 75
HP Secure Sockets Layer (SSL) for OpenVMS 76
HP TCP/IP Services for OpenVMS 76

6. OpenVMS Documentation Overview

7. OpenVMS Printed and Online Documentation

Printed Documentation 83
OpenVMS Media Kit Documentation 84
OpenVMS Documentation Sets 84
Documentation for System Integrated Products 87
Archived OpenVMS Documentation 87
New Authoring Tool for OpenVMS Documentation 88
Online Documentation CD-ROM 88
Online Formats 89
PDF Reader 89
Online Documentation on the OpenVMS Web Site 89
Online Help 89

8. Descriptions of OpenVMS Manuals

Manuals in the OpenVMS Media Kit 91
Manuals in the Base Documentation Set 92
Additional Manuals in the OpenVMS Full Documentation Set 93
RMS Journaling Manual 97
Archived Manuals 97

Index 105

Contents

Table 1-1. Summary of General User Features 19
Table 1-2. Summary of System Management Features 19
Table 1-3. Summary of Programming Features 22
Table 1-4. Summary of Associated Products New Features 23
Table 2-1. Command Size Limits. 27
Table 2-2. Updates to DCL Commands and DCL Documentation 28
Table 2-3. Updates to DCL Lexical Functions and Lexicals Documentation 30
Table 3-1. BACKUP Option Structure Types 35
Table 3-2. BCK_OPT_K_UNSHELVE. 35
Table 3-3. LAN Failover and failSAFE IP Comparison 46
Table 4-1. SDA Commands for the OCLA Utility. 67
Table 4-2. New and Revised System Services. 69
Table 5-1. Supported Video Modes 72
Table 6-1. Documentation Set Changes for OpenVMS Alpha Version 7.3-2. 81
Table 7-1. OpenVMS Media Kit Documentation 84
Table 7-2. OpenVMS Documentation Sets 84
Table 7-3. OpenVMS Full Documentation Set (QA-001AA-GZ.7.3-2). 85
Table 7-4. System Integrated Products Documentation 87
Table 8-1. Archived OpenVMS Manuals. 97
Table 8-2. Archived Networking Manuals and Installation Supplements 99

Tables

Figure 4-1. DDT Intercepts 58

Figures

Preface

The *HP OpenVMS Alpha Version 7.3-2 New Features and Documentation Overview* describes how customers can take advantage of and find information about the new features available in OpenVMS Alpha Version 7.3-2.

The information in this document applies to OpenVMS Alpha systems only; it does not apply to OpenVMS VAX systems.

Intended Audience

This guide is intended for system managers, application programmers, technical consultants, data center managers, and anyone else who wants to learn about the new features in OpenVMS Alpha Version 7.3-2.

Document Structure

This guide introduces OpenVMS some_topic concepts and features available in OpenVMS Alpha Version 7.3-2.

Part I, OpenVMS Alpha Version 7.3-2 New Features

- Chapter 1 summarizes the new OpenVMS software features.
- Chapter 2 describes the new features of interest to general users of the OpenVMS Alpha operating system.
- Chapter 3 describes the new features that pertain to the tasks performed by system managers.
- Chapter 4 describes the new features that support programming tasks.
- Chapter 5 describes new features for associated products.

Part II, OpenVMS Documentation

- Chapter 6 describes the OpenVMS documentation changes from the previous version.
- Chapter 7 describes how the documentation is delivered.
- Chapter 8 describes each manual in the OpenVMS documentation set.

Related Documents

For additional information about HP OpenVMS products and services, see the following World Wide Web address:

`http://www.hp.com/products/openvms`

Reader's Comments

HP welcomes your comments on this manual.

Please send comments to either of the following addresses:

Internet: openvmsdoc@hp.com

Postal Mail:
Hewlett-Packard Company
OSSG Documentation Group
ZKO3-4/U08
110 Spit Brook Road
Nashua, NH 03062-2698

How to Order Additional Documentation

For information about how to order additional documentation, visit the following World Wide Web address:

`http://www.hp.com/go/openvms/doc/order/`

Conventions

The following conventions may be used in this manual:

Convention	Meaning
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	A sequence such as PF1 *x* indicates that you must first press and release the key labeled PF1 and then press and release another key (x) or a pointing device button.
Return	In examples, a key name in bold indicates that you press that key.
...	A horizontal ellipsis in examples indicates 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.
. . .	A vertical ellipsis indicates the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.
( )	In command format descriptions, parentheses indicate that you must enclose choices in parentheses if you specify more than one.
[ ]	In command format descriptions, brackets indicate optional choices. You can choose one or more items or no items. Do not type the brackets on the command line. However, you must include the brackets in the syntax for OpenVMS directory specifications and for a substring specification in an assignment statement.
|	In command format descriptions, vertical bars separate choices within brackets or braces. Within brackets, the choices are optional; within braces, at least one choice is required. Do not type the vertical bars on the command line.
{ }	In command format descriptions, braces indicate required choices; you must choose at least one of the items listed. Do not type the braces on the command line.

Convention	Meaning
bold type	**Bold type** represents the introduction of a new term. It also represents the name of an argument, an attribute, or a reason. In command or script examples, bold text indicates user input.
italic type	Italic type 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 *dd* represents the predefined par code for the device type).
UPPERCASE TYPE	Uppercase type indicates a command, the name of a routine, the name of a file, or the abbreviation for a system privilege.
`Example`	This typeface indicates code examples, command examples, and interactive screen displays. In text, this type also identifies URLs, UNIX command and pathnames, PC-based commands and folders, and certain elements of the C programming language.
–	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.

I OpenVMS Alpha Version 7.3-2 New Features

This part contains information about new features in OpenVMS Alpha Version 7.3-2.

1 Summary of HP OpenVMS Alpha 7.3-2 New Features

HP OpenVMS Alpha Version 7.3-2 delivers the highest levels of availability, scalability, flexibility, performance, and security that are required for operating in a 24x365 environment. With more than 20 years of proven reliability, OpenVMS continues to enhance its availability and performance by including new technology in the base operating system and in the HP OpenVMS Cluster software environment.

OpenVMS Alpha Version 7.3-2 focuses on expanding our OpenVMS Storage Area Network (SAN) offering, improving system availability, providing I/O and SMP performance improvements, and enhanced security. This release also includes enhancements that make it easier to port UNIX® applications to OpenVMS.

Performance Features

OpenVMS Alpha 7.3-2 focuses on making operating system changes that improve system, SMP, and I/O performance, with the ultimate goal of improving customer application performance. OpenVMS Alpha Version 7.3-2 includes a number of performance enhancements across a range of areas:

- C Run-Time Library enhancements
- COPY and SEARCH performance enhancements
- Fast Path for LAN drivers
- $GETRMI system service
- Large pool lookaside lists
- PEdriver Fast Path
- Fast Path for the Smart Array 53*xx* series adapter (KZPDC)

Security Features

The following security features in OpenVMS Alpha Version 7.3-2 provide flexibility and simplified software development options for authentication and cryptography:

- Authentication and Credentials Management Extensions (ACME) subsystem
- Common Data Security Architecture (CDSA) Version 2.0
- HP Kerberos Version 2.0 for OpenVMS
- HP Secure Sockets Layer (SSL) for OpenVMS

This manual provides an overview of each security feature as well as references for more information.

UNIX Portability Features

In addition to providing an Open Source Tools CD-ROM that contains software tools to help you port UNIX applications to OpenVMS Alpha, Version 7.3-2 of the operating system includes internal enhancements to support the porting process and multiplatform environments, including:

- Improved overall security features of the operating system.
- Improved UNIX style security features that allow for ease of porting UNIX style applications to an OpenVMS system.

System Management Features

This manual provides an overview of the following system management features included in OpenVMS Alpha Version 7.3-2:

- AlphaServer system DS15 support
- Alpha Server Series ES47, ES80, and GS1280 system support
- Changes to the Backup utility
- Component indictment
- Dissimilar device shadowing (DDS) and dynamic volume expansion (DVE)
- Error Log Viewer (ELV) utility
- Enhanced font support
- Extended File Cache (XFC) enhancements
- New FIBRE_SCAN utility for displaying device information
- Mount Verify message reduction
- POLYCENTER Software Installation utility
- SET DEVICE/RESET command
- HP StorageWorks Modular SAN array 1000 support (MSA 1000)
- SYSMAN DUMP_PRIORITY commands
- UNIX portability security features
- USB Configuration Manager (UCM) utility
- HP Volume Shadowing for OpenVMS management functions

Summary Tables

OpenVMS Alpha Version 7.3-2 provides all the capabilities of OpenVMS Version 7.3-1 plus the new features added to the OpenVMS Alpha operating system. Table 1-1, Table 1-2, Table 1-3, and Table 1-4 summarize the new features in Version 7.3-2 according to their functional area.

Table 1-1 Summary of General User Features

Feature	Description
COPY and SEARCH performance enhancements	The performance of the DCL commands COPY and SEARCH have been enhanced.
Blocks-to-bytes display option	Some DCL command functions are modified to allow users to display size information in bytes instead of blocks.
DCL command size limits increased	Extended DCL (EDCL) increases the command size limits of the command line interpreter (CLI).
DCL commands and lexical functions	Numerous enhancements are added to DCL commands and lexical functions.
Documentation updates	Updated OpenVMS documentation is available on the World Wide Web.
MIME utility enhancements	Several enhancements were made to this utility.

Table 1-2 Summary of System Management Features

Feature	Description
AlphaServer system DS15 support	Includes support for the new DS15 system.
AlphaServer Series systems ES47, ES80, and GS1280 support	Includes support for these systems without a supplementary kit, as well as Galaxy and hard partitions.
New BACKUP command qualifiers	Allow you to specify a list of files as input to a BACKUP command and to specify when a save set is restored. Also allow you to save only a file header in a BACKUP operation.
Component indictment	Provides support of component indictment of CPUs.
Dissimilar device shadowing (DDS)	Supports the shadowing of devices of different sizes, to be used with dynamic volume expansion.
Dynamic volume expansion (DVE)	Enables you to increase the logical volume size (the amount of disk space added to the file system) without dismounting the device.

Table 1-2 Summary of System Management Features (Continued)

Feature	Description
Enhanced font support	Includes support for the euro symbol, along with additional, fixed-width and scalable fonts.
Error Log Viewer (ELV) utility	Allows you to quickly examine, from the command line, an error log file in a user-readable format before deciding whether the data warrants a more comprehensive analysis with a tool such as the System Event Analyzer.
Extended File Cache (XFC) enhancements	Includes performance and usability improvements to XFC.
Fast Path for LAN devices	Enhancements help streamline I/O processing and improve SMP performance scalability.
Fast Path support for the Smart Array 53*xx* Series adapter (KZPDC)	Provides Fast Path support for the KZPDC adapter.
FIBRE_SCAN utility	Displays information about all storage devices attached to Fibre Channel on the current system; both configured and non-configured devices are included.
Gigabit Ethernet adapter (DEGXA) support	Includes support for DEGXA in OpenVMS Alpha Version 7.3-2.
Installation, upgrade, and configuration procedures	Simplified procedures reduce the number of options to choose.
LAN Failover	Provides a mechanism to protect your system against a network interface card (NIC) failure.
Mail flag	MAIL$SYSTEM_FLAGS added to the Mail utility.
Mount Verify message reduction	Mount verification messages for infrequent, immediately recovered events are suppressed from operator logs. Two system parameters are available for logging these messages.
PEdriver Fast Path support	Improves SMP performance and provides other enhancements.

Table 1-2 Summary of System Management Features (Continued)

Feature	Description
POLYCENTER Software Installation utility	• The /RECOVER_MODE qualifier protects your system from a PRODUCT INSTALL or PRODUCT RECONFIGURE command failure by automatically restoring the system environment. • The /SAVE_RECOVERY_DATA qualifier for the PRODUCT INSTALL command, in conjunction with three new PRODUCT commands, allows you to uninstall patches and manage the associated recovery data. • You can create and install product kits in compressed format. • The PRODUCT SHOW OBJECT command now displays file and module generation numbers.
SET DEVICE/RESET command	The SET DEVICE command can now reset device counts.
System parameters	The SHADOW_REC_DLY parameter represents the number of seconds a system waits before it attempts to manage transient- state operations on any virtual units that are mounted on the system. The SHADOW_SITE_ID parameter allows a system manager to define a site value, which the OpenVMS volume shadowing software uses to determine the best device to perform reads, thereby improving performance.
StorageWorks Modular SAN array 1000 support (MSA 1000)	New high-performance Fibre Channel storage array. In the future, the MSA1000 will enable a two-node cluster in an arbitrated loop configuration by means of its embedded three-port Fibre Channel arbitrated loop hub. Support for this configuration will be provided in a patch kit.
SYSMAN DUMP_PRIORITY commands	Allow you to increase the probability that key processes will fit into a selective dump.
UNIX portability security features	Support for GUID and UID security identifiers, mixed-case passwords, and new screen-lock and idle-process killers.

Table 1-2 Summary of System Management Features (Continued)

Feature	Description
USB Configuration Manager (UCM) utility	The new USB (Universal Serial Bus) Configuration Manager (UCM) utility allows you to connect a computer to a variety of devices using a single four-wire cable.
Volume Shadowing for OpenVMS	New SET SHADOW, SHOW SHADOW, and ANALYZE/DISK/SHADOW management functions allow greater management control over shadowed volumes.

Table 1-3 Summary of Programming Features

Feature	Description
Authentication and Credentials Management Extensions (ACME)	Provides authentication and persona-based credential services.
HP C Run-Time Library enhancements	Enhancements provide improved UNIX portability and the flexibility of selecting additional user-controlled features. New C RTL functions are also included.
DECdtm/XA Version 2.1	Provides clusterwide transaction recovery support. Thus, transactions from applications that use the clusterwide DECdtm Gateway Domain Log can now be recovered from any single-node failure.
$GETDVI item code DVI$_SHDW_SITE	Returns to the user the SITE value used by shadowing.
$GETRMI system service	Returns system performance information about the local system, increased from 37 to 258 item codes.
Large-pool lookaside lists	Enhancements increase the lookaside list size.
SCSI and Fibre Channel Multipath support for third-party applications	New routines allow modification of third-party SCSI disk-caching or disk-shadowing applications to run in an OpenVMS SCSI or Fibre Channel multipath configuration.
System Dump Analyzer (SDA) commands	New commands are provided for the Alpha EV7 On Chip Logic Analyzer utility.

Table 1-4 Summary of Associated Products New Features

Feature	Description
ATI RADEON 7500 graphics support	Software support for the ATI RADEON 7500 graphics card.
Common Data Security Architecture (CDSA) V2.0	This port of CDSA V2 Release 3.14 to OpenVMS provides support for all core CDSA functions. CDSA V2.0 ships with OpenVMS Alpha Version 7.3-2 and is automatically installed with the operating system.
OpenVMS e-Business Infrastructure Package	Several components of the V1.5 package have been updated.
Kerberos V2.0 for OpenVMS	Now included as a layered product. New features include a fix for the security vulnerability, support for triple DES encryption, database enhancements, DNS support for locating Key Distribution Centers, and Kerberos 4.0 interoperability.
Secure Sockets Layer (SSL) V1.1-A for OpenVMS	Enhancements include the OpenSSL 0.9.6G baselevel, Certificate Revocation List (CRL) support, a Data Encryption Standard (DES) encryption image.

2 General User Features

This chapter provides the following information about new features for all users of the OpenVMS operating system:

- Blocks-to-bytes display option
- COPY and SEARCH performance enhancements
- DCL command size limits increased
- DCL commands and lexical functions
- Documentation updates on the World Wide Web
- MIME utility enhancements

Blocks-to-Bytes Display Option

The following DCL command functions have been modified to allow users to display size information in bytes instead of blocks:

COPY
DELETE
DIRECTORY
PURGE
SHOW DEVICES
SHOW DEVICES/SERVED
SHOW MEMORY
SHOW QUOTA

To change your default display from blocks to bytes for all functions, enter the DCL command SET PROCESS/UNITS=BYTES. The SHOW PROCESS/UNITS command displays your current default. You can override the current default for individual DIRECTORY, SHOW DEVICES, and SHOW MEMORY commands by using the following qualifiers, as appropriate:

DIRECTORY/SIZE=UNITS=BLOCKS
DIRECTORY/SIZE=UNITS=BYTES
SHOW DEVICES/UNITS=BLOCKS
SHOW DEVICES/UNITS=BYTES
SHOW MEMORY/UNITS=BLOCKS
SHOW MEMORY/UNITS=BYTES
SHOW MEMORY/UNITS=PAGES

Example 2-1 shows the output of the blocks-to-bytes display option:

Example 2-1 Block-to-Bytes Display Option

```
$ SHOW PROCESS/UNITS
```

```
22-MAY-2003 12:50:20.13     User:   SYSTEM          Process ID:        24C00125
                             Node:  TEST            Process name:      "SYSTEM"

Soft CPU Affinity: off

Units:  Blocks

$ DIRECTORY *.PS

Directory $1$DKC600:[TEST]

REPORT.PS;1                    1197

Total of 1 file,  1197 blocks.

$ DIRECTORY/SIZE=UNITS=BYTES *.PS

Directory $1$DKC600:[TEST]

REPORT.PS;1                    598KB

Total of 1 file,  598KB

$ SET PROCESS/UNITS=BYTES
$ DIRECTORY *.PS

Directory $1$DKC600:[TEST]

REPORT.PS;1                    598KB

Total of 1 file,  598KB
$
```

The units property has been added to the system service $SET_PROCESS_PROPERTIES. There is also a new run-time library routine (LIB$CVT_BLOCKS_BYTES) that performs blocks-to-bytes conversions.

COPY and SEARCH Performance Enhancements

Performance of the DCL commands COPY and SEARCH has been significantly improved. It now takes less time and fewer computing resources to copy a file or perform a search.

DCL Command Size Limits Increased

Extended DCL (EDCL) increases the command size limits of the command line interpreter (CLI). EDCL brings the command line size in line with other operating system CLIs. The main goals of EDCL are to ease the porting of software to the OpenVMS platform and to ease the use of long file specifications in DCL commands and command procedures.

Table 2-1 lists the increased size limits for certain structures in OpenVMS Alpha Version 7.3-2.

Table 2-1 Command Size Limits

Structure	Old Size Limits	New Size Limits
Interactive DCL commands	255 bytes	4095 bytes
DCL commands in a file	1024 bytes	8192 bytes
DCL symbols	1024 bytes	8192 bytes

The following components were also modified to support the size increase in the command line buffer:

CRTL (argv, argc)
DCL commands READ and WRITE
Foreign commands
Lexical functions
LIB$DO_COMMAND
LIB$GET_COMMAND
LIB$GET_FOREIGN
LIB$GET_SYMBOL
LIB$SET_SYMBOL
Recall buffer

Unchanged Size Limits

The size of logical names has not changed. A logical name cannot store a file specification or string that exceeds 255 bytes. Symbols must be used to pass long strings in DCL or between the CLI and the application.

The token size limit also remains unchanged at 255 bytes. This size may be increased in a future release.

DCL Commands and Lexical Functions

This section lists new and changed DCL commands, qualifiers, and lexical functions for the OpenVMS Alpha Version 7.3-2 operating system. Table 2-2 and Table 2-3 summarize these changes. See “POLYCENTER Software Installation New Features” on page 47 for information on the new PRODUCT commands and qualifiers.

For more information, refer to online help or to the *HP OpenVMS DCL Dictionary* unless otherwise noted.

Table 2-2 lists updates to DCL commands and DCL documentation.

Table 2-2 Updates to DCL Commands and DCL Documentation

DCL Command	Change or Update
ANALYZE/ERROR_LOG	This new command has been added. See the online help or the *HP OpenVMS System Management Utilities Reference Manual* for details.
ANALYZE/ERROR_LOG/ELV	This new command has been added. See the online help or the *HP OpenVMS System Management Utilities Reference Manual* for details.
ANALYZE/IMAGE	The /SELECT=IDENTIFICATION qualifier has been changed to /SELECT=IDENTIFICATION=*keyword*.
CLOSE	The /DISPOSITION qualifier has been added.
CREATE	A new example has been added.
DEBUG	The /TARGET_ARCHITECTURE qualifier has been added for DEBUG/KEEP.
DIRECTORY	The UNITS option has been added to the /SIZE qualifier, and the UNUSED option has been added to the /SELECT=SIZE qualifier.
DISMOUNT	The /POLICY=MINICOPY qualifier requires the LOG_IO privilege to create bitmaps.
INITIALIZE	The /LIMIT and /VOLUME_CHARACTERISTICS qualifiers have been added. The /CLUSTER_SIZE, /EXTENSION, /HEADERS, /HIGHWATER, and /HOMEBLOCKS qualifiers have been updated for ODS-5 disks.
LBXPROXY [1]	This new command provides support for the Low-Bandwidth X (LBX) proxy server. For information about using LBXPROXY, see the *HP DECwindows Motif for OpenVMS Alpha New Features* manual.
LOGIN	The /CONNECT and /LOCAL_PASSWORD qualifiers have been added.
MOUNT	The /POLICY_MINICOPY qualifier requires the LOG_IO privilege to create bitmaps.
PIPE	The description has been changed to clarify that PIPE always returns the status of the command performed in the last segment.
RECALL	The /SEARCH qualifier has been added, and the /ALL qualifier now accepts a parameter.
RENAME	A new example has been added.
SET CACHE/RESET	A note that this command requires the OPER privilege has been added.
SET CPU	The /ASSIGN qualifier has been added, and the /FAILOVER and /MIGRATE qualifiers have been updated.
SET DEFAULT	A clarifying note has been added to the description of device-name.

Table 2-2 Updates to DCL Commands and DCL Documentation (Continued)

DCL Command	Change or Update
SET DEVICE	The /MV_INTVL, /MV_NUM, and /RESET qualifiers have been added; descriptions of the /PATH, /READ_COST, and /SITE qualifiers have been updated.
SET DISPLAY	The following new qualifiers have been added to provide the ability to define named properties, as well as to support the X Authority utility (xauth), the Low-Bandwidth X (LBX) proxy server and proxy manager application: /GENERATE, /[NO]LBXAUTHENTICATE, /LBXDATA, /PMPORT, /PMTRANSPORT, /PROXY, /QUOTA, /REVOKE, /VALUE, and /[NO]XAUTHORITY. The command description has been updated to describe these new capabilities, and new examples have been provided. The descriptions for the /SERVER and /TRANSPORT qualifiers have also been updated to describe their use with a proxy manager or standalone proxy server.
SET NETWORK	An example has been added and old examples have been updated.
SET PASSWORD	The specifications for user passwords have been expanded for users with PWDMIX authorization.
SET PROCESS	The /CASE_LOOKUP and /UNITS qualifiers have been added.
SET SHADOW	This new command has been added.
SET VOLUME	The /LIMIT, /SIZE, and /VOLUME_CHARACTERISTICS qualifiers have been added, and the description of /LABEL has been clarified.
SHOW CPU	New examples have been added.
SHOW DEVICES	The /UNITS qualifier has been added, and the usage of the /SIZE qualifier has been changed.
SHOW DISPLAY	The /ALL, /EXTRACT, /QUOTA, /SYMBOLS, and /VALUES qualifiers have been added.
SHOW MEMORY	The /UNITS qualifier has been added, and the description of the /FULL qualifier has been revised.
SHOW PROCESS	The /SCHED and /UNITS qualifiers have been added.
SHOW SHADOW	This new command has been added.
SHOW SYSTEM	The /[NO]TOTAL and the /[NO]GRAND_TOTAL qualifiers have been added, along with a new example.
STOP	The description section has been clarified.
STOP/CPU	The /ASSIGN qualifier has been added, along with two new examples.
XAUTH	This new command invokes the X Authority utility (xauth) used to manage the content of one or more X authority files, which control access to an X display server. For more information about using xauth, see the *HP DECwindows Motif for OpenVMS Alpha New Features* manual.

Table 2-2 Updates to DCL Commands and DCL Documentation (Continued)

DCL Command	Change or Update
XPROXYMANAGER [1]	This new command provides support for the proxy manager application used to control Low-Bandwidth X (LBX) proxy server configurations. For information about using XPROXYMANAGER, see the *HP DECwindows Motif for OpenVMS Alpha New Features* manual.

[1] This command is available only with the installation of the Low-Bandwidth X support, which is an optional component of the DECwindows Motif layered product.

Table 2-3 lists updates to DCL lexical functions and lexicals documentation.

Table 2-3 Updates to DCL Lexical Functions and Lexicals Documentation

DCL Lexical Functions	Documentation Update
F$DELTA_TIME	This new lexical function calculates the elapsed time between a start and end time.
F$GETDVI	The following new items have been added: EXPSIZE, QLEN, SHDW_COPIER_NODE, SHDW_DEVICE_COUNT, SHDW_GENERATION, SHDW_MASTER_MBR, SHDW_MBR_COPY_DONE, SHDW_MBR_COUNT, SHDW_MBR_MERGE_DONE, SHDW_MBR_READ_COST, SHDW_MINIMERGE_ENABLE, SHDW_READ_SOURCE, SHDW_SITE, SHDW_TIMEOUT, VOLCHAR, and VOLSIZE. The MEDIA_ID and PREFERRED_CPU items have been corrected to say that they are returned as integers.
F$GETJPI	CASE_LOOKUP_IMAGE, CASE_LOCKUP_PERM, and ORG_BYTLM have been added. The MSGMASK item text has been updated, and a new example has been added.
F$GETQUI	Two new items, FILE_DEVICE and FILE_DID, have been added; the FILE_IDENTIFICATION item description has been updated; the DISPLAY_ENTRY function has been added to the THIS_JOB keyword.
F$GETSYI	FREE_PAGES, MODIFIED_PAGES, TOTAL_PAGES, and USED _PAGES have been added, and the description of RAD_MAX_RADS has been modified. ARCHFLAG and REAL_CPUTYPE have been corrected to say they are returned as integers.
F$UNIQUE	This new lexical function has been added. It returns a different string each time it is called.

Documentation Updates on the World Wide Web

In an effort to provide timely updates for our customers, OpenVMS posts new, supplemental, and updated information to all OpenVMS product documentation sets on the World Wide Web. We encourage customers to check the OpenVMS Documentation web site for changes to the OpenVMS and layered product manuals, as well as pertinent papers and other documents of interest to our customers. Documents posted on the site are clearly marked and include the revision date.

The OpenVMS Documentation web site is located at the following URL:

`http://www.hp.com/go/openvms/doc/`

MIME Utility Enhancements

Version 1.8 of the MIME utility includes the following new features:

- Added support for the MAIL$EDIT logical name to select an editor for use with the MIME NEW and EDIT commands. You can also specify a procedural script using one of the following command lines:

  ```
  $ define MAIL$EDIT callable_lse
  ```

  ```
  $ define MAIL$EDIT "MIME_EDIT_SCRIPT.COM"
  ```

- Added support for the /EDIT (/NOEDIT) qualifier for use with the NEW and EDIT commands. You can also specify a procedural script using one of the following command lines:

  ```
  /EDIT="edit/tpu"
  ```

  ```
  /EDIT="@MIME_EDIT_SCRIPT.COM"
  ```

3 System Management Features

This chapter provides the following information about new features, changes, and enhancements for system management:

- AlphaServer systems support
- Backup utility changes and additions
- Component indictment
- Dissimilar device shadowing (DDS) and dynamic volume expansion (DVE)
- Error Log Viewer (ELV) utility
- Euro symbol and additional fixed-width and scalable font sets support
- Extended File Cache (XFC) enhancements
- Fast Path for LAN devices
- Fast Path support for Smart Array 53*xx* Series Adapter (KZPDC)
- Fibre Channel new features
- Gigabit Ethernet adapter (DEGXA) support
- Installation, upgrade, and configuration procedures
- LAN Failover
- Mail flag
- Mount Verify message reduction
- POLYCENTER Software Installation utility new features
- SET DEVICE/RESET command
- System parameters
- SYSMAN DUMP_PRIORITY commands
- UNIX portability security features
- USB Configuration Manager (UCM) utility
- Volume Shadowing for OpenVMS management functions

AlphaServer Systems

The following sections describe the AlphaServer systems that are now supported by the OpenVMS Alpha Version 7.3-2 operating system.

AlphaServer System DS15 Support

OpenVMS Alpha Version 7.3-2 supports the new AlphaServer system DS15, which was introduced in September 2003, and replaces the AlphaServer system DS10. OpenVMS Alpha Version 7.3-2 supports this system without a supplementary kit. (Note that the selected software or hardware components my require an update kit, as described in the *HP OpenVMS Alpha Version 7.3-2 Release Notes*.)

For more information about the DS15 system, refer to the QuickSpec at:

`http://h18003.www1.hp.com/alphaserver/`

AlphaServer Series Systems ES47, ES80, and GS1280 Support

The newest members of the HP AlphaServer Series systems, the ES47 and GS1280, were introduced in January, 2003, followed in April, 2003, by the ES80 Series. The state-of-the-art hardware architecture of these new AlphaServer systems, including the mesh architecture and low latency I/O subsystem of the EV7 processor, provide customers with robust and cost-effective systems. Running OpenVMS Alpha, these new systems offer outstanding performance, scalability, and availability.

OpenVMS Alpha Version 7.3-2 is the first OpenVMS release that supports these systems without a supplementary kit. (Note that the selected software or hardware components may require an update kit, as described in the *HP OpenVMS Alpha Version 7.3-2 Release Notes*.)

Galaxy (soft partitions) and hard partitions are supported on these systems.

For more information about these systems, refer to the QuickSpecs for each system series at:

`http://h18003.www1.hp.com/alphaserver/`

For more information about Galaxy and hard partitions, refer to the *HP OpenVMS Alpha Partitioning and Galaxy Guide*. For future Galaxy support announcements, see OpenVMS web site:

`http://www.hp.com/go/openvms`

Backup Utility Changes and Additions

The following sections provide additions and changes to the Backup utility and a new example.

New BACKUP Command Qualifiers

The following new qualifiers for the BACKUP command have been added in OpenVMS Version 7.3-2:

- /FILE_SELECTED – allows you to specify a file that contains a list of the files that are to be selected when a save set is restored. You can use this qualifier in place of the /SELECT qualifier to select files to restore from a save set.
- /HEADER_ONLY – specifies that only the file headers of a file are to be saved in a BACKUP operation (rather than unshelving shelved files before backing them up, which is the default behavior).
- /INPUT_FILES – allows you to specify a list of files to be processed for input to a BACKUP command.

More information is available in the BACKUP chapter of the *HP OpenVMS System Management Utilities Reference Manual*.

New BACKUP Option Structure Types

Table 3-1 lists the new option structure types that have been added to the Backup utility.

Table 3-1 BACKUP Option Structure Types

Option Structure	Description
BCK_OPT_K_FILES_SELECTED_IN	32-bit descriptor specifying the name of a file that contains the names of files to be restored from a saveset. **NOTE:** Use of BCK_OPT_K_FILES_SELECTED_IN and BCK_OPT_K_SELECT in the same call results in a fatal error.
BCK_OPT_K_HSMHEADER_ONLY	Flag reserved for use with HSM product.
BCK_OPT_K_HSMOPTIONS_IN	Flag reserved for use with the HSM product.
BCK_OPT_K_HSMOPTIONS_OUT	Flag reserved for use with the HSM product.
BCK_OPT_K_INPUT_FILES_IN	Flag specifying that the file name descriptor in BCK_OPT_K_INPUT is to be used as a list of files for input.
BCK_OPT_K_PRESHELVED	Flag used with the file-shelving layered products. The values are: TRUE - Save the header and data of a preshelved file (default). FALSE - Save only the header of a preshelved file. **NOTE:** Use of BCK_OPT_K_PRESHELVED and BCK_OPT_K_PHYSICAL in the same call results in a fatal error.

Change to BCK_OPT_K_UNSHELVE

The description of BCK_OPT_K_UNSHELVE in Table 3-2 replaces the description in *OpenVMS Utility Routines Manual*, Table 3-2.

Table 3-2 BCK_OPT_K_UNSHELVE

Option Structure	Description
BCK_OPT_K_UNSHELVE	Flag used with the file-shelving layered products. The values are: TRUE - Save the header and data of a shelved file (default). FALSE - Save only the header of a shelved file. **NOTE:** Use of BCK_OPT_K_UNSHELVE and BCK_OPT_K_PHYSICAL in the same call results in a fatal error.

New BACKUP API Example

The C example program in Example 3-1 demonstrates calling the BACKUP API that performs the following DCL command:

```
$ BACKUP FILE.DAT/INPUT_FILES A.BCK/SAVE_SET
```

Example 3-1 Example BACKUP API Program

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ssdef.h>
#include <descrip.h>
#include "sys$examples:bapidef.h"
/*
** Macros to fill in the BACKUP structure
*/
/*
** Add a flag to the BACKUP structure
**   _t = option_type
**   _v = opt_flag_value
**   _p = pointer to memory -- where to copy the flag_structure
**        and then increment it by bck_opt_struct_flag amount
*/
#define ADD_FLAG(_t,_v,_p) \
  { \
    bck_opt_struct_flag flag_struct; \
    flag_struct.option_type = _t; \
    flag_struct.opt_flag_value = _v; \
    memcpy (_p, &flag_struct, sizeof( bck_opt_struct_flag )); \
    _p = _p + sizeof( bck_opt_struct_flag ); \
  }
/*
** Add a descriptor to the BACKUP structure
**   _t = option_type
**   _s = string to add
**   _p = pointer to memory -- where to copy the descriptor_structure
**        and then increment it by bck_opt_struct_dsc amount
*/
#define ADD_DSC(_t,_s,_p) \
  { \
    bck_opt_struct_dsc dsc_struct; \
    dsc_struct.dsc_flags = 0;\
    dsc_struct.opt_dsc_type = _t; \
    dsc_struct.opt_dsc.dsc$b_dtype = DSC$K_DTYPE_T; \
    dsc_struct.opt_dsc.dsc$b_class = DSC$K_CLASS_S; \
    dsc_struct.opt_dsc.dsc$w_length = strlen(_s); \
    dsc_struct.opt_dsc.dsc$a_pointer = _s; \
    memcpy (_p, &dsc_struct, sizeof( bck_opt_struct_dsc )); \
    _p = _p + sizeof( bck_opt_struct_dsc ); \
  }
extern unsigned int backup$start (char   *par_arg_buff );
main ()
```

```
{
  char      bck_buf[1000];
  char      *bck_buf_ptr = &bck_buf[0];
  char      input_str[100] = "FILE.DAT";
  char      output_str[100] = "A.BCK";
  unsigned long int   status;


  ADD_DSC( BCK_OPT_K_INPUT, input_str, bck_buf_ptr );
  ADD_FLAG( BCK_OPT_K_OPERATION_TYPE, BCK_OP_K_SAVE, bck_buf_ptr );
  ADD_FLAG( BCK_OPT_K_INPUT_FILES_IN, TRUE, bck_buf_ptr );
  ADD_FLAG( BCK_OPT_K_SAVE_SET_OUT, TRUE, bck_buf_ptr );
  ADD_FLAG( BCK_OPT_K_ASSIST, TRUE, bck_buf_ptr );
  ADD_DSC( BCK_OPT_K_OUTPUT, output_str, bck_buf_ptr );


  /*
  **  Indicate the end of options that specify the BACKUP operation
  **  to be performed
  */
  ADD_FLAG( BCK_OPT_K_END_OPT, FALSE, bck_buf_ptr );


  status = backup$start( &bck_buf[0] );
  if (!( status & 1))
  {
   exit ( status ); // Exit if the first part of the test failed
  }
  exit( SS$_NORMAL );
}
```

Component Indictment

Component indictment, a new feature in OpenVMS AlphaVersion 7.3-2, provides higher system availability by stopping a failing CPU before it can cause a system halt. The component indictment process uses the fault-analysis utility System Event Analyzer (SEA; formerly known as Compaq Analyze) to identify CPUs that are incurring a high or abnormal incidence of correctable errors. System Event Analyzer notifies OpenVMS if a CPU is failing. OpenVMS can then remove the CPU from the active set before it causes the system to fail.

The following requirements are needed to support component indictment:

- One of the following AlphaServer systems: ES45, DS25, GS80, GS160, GS320, ES47, ES80, or GS1280
- OpenVMS Alpha Version 7.3-2 installed
- System Event Analyzer (SEA) Version 4.2 (or higher) *installed and running in automatic analysis mode*
- A properly edited or created SYS$MANAGER:SYS$INDICTMENT_POLICY.COM file

The SYS$MANAGER:SYS$INDICTMENT_POLICY.COM file defines the policy on your particular system for CPU indictment. The default policy is to not allow any CPUs to be stopped. In this instance, the policy command procedure would have declared the logical name SYS$INDICT_ALLOW_CPUS equal to 0, or, more precisely, not defined to 1. If you want to support the stopping of all indicted CPUs, edit the policy command procedure using the following steps:

1. At the DCL prompt, change the SYS$INDICT_START logical name (located in the SYS$INDICTMENT_POLICY.COM file) to equal 1. For example:

   ```
   $ DEFINE/SYSTEM/EXECUTIVE_MODE/NOLOG SYS$INDICT_START 1
   ```

2. Change the SYS$INDICT_ALLOW_CPUS logical name (located in the SYS$INDICTMENT_POLICY.COM file) to equal 1. For example:

   ```
   $ DEFINE/SYSTEM/EXECUTIVE_MODE/NOLOG SYS$INDICT_ALLOW_CPUS 1
   ```

You can develop more complex policies for your particular system. For more information regarding policy creation, refer to the SYS$INDICTMENT_POLICY.COM file. This file must be maintained correctly for component indictment to function.

Overview of Indictment Process

The component indictment process follows this order:

1. A CPU begins exhibiting correctable errors, which are entered into the error log.
2. System Event Analyzer (SEA) reads the error log and performs an analysis. If the analysis concludes that the component may have an unrecoverable error, then SEA notifies the OpenVMS operating system that the component should be considered for replacement by issuing an indictment request.
3. When the OpenVMS operating system receives the indictment, it checks the logical definitions defined by the COM file in the SYS$INDICTMENT_POLICY.COM file to determine whether the CPU should be stopped. If the policy allows it, the CPU is stopped (the equivalent of issuing the STOP/CPU command on the CPU).

   When the system is rebooted, the CPU will be in the Active Set again.

Sample SYS$INDICTMENT_POLICY.COM

The following is a sample from the a SYS$INDICTMENT_POLICY.COM file:

```
$!
$!SYS$INDICTMENT_POLICY.COM
$!
$!This command procedure enables a system manager to:
$!
$!a. Enable or disable the starting of the indictment server.
$!
$!b. Set policy to enable or disable an attempt by the operating system to
$! STOP a CPU that has been indicted.
$!
$!Setting the definition of SYS$INDICT_START to 1 enables
$!the indictment server to be started; setting it to 0 disables the server.
$!By default, the indictment server is not started.
$!The following definition disables the server:
$!
```

```
$DEFINE/SYSTEM/EXECUTIVE_MODE/NOLOG SYS$INDICT_START 0
$!
$!Setting the definition of SYS$INDICT_ALLOW_CPUS to 1 enables
$!the CPUs to stop; setting it to 0 disables the stopping of CPUs.
$!By default, CPUs are not stopped.
$!The following definition disables CPUs from being stopped.
$!
$DEFINE/SYSTEM/EXECUTIVE_MODE/NOLOG SYS$INDICT_ALLOW_CPUS 0
$!
$!The system manager can change the SYS$INDICT_ALLOW_CPUS logical name
$!at any time to control the stopping of CPUs.
$!
$EXIT
```

Dissimilar Device Shadowing and Dynamic Volume Expansion

OpenVMS Alpha Version 7.3-2 supports the shadowing of dissimilar devices that differ in size. This functionality is called dissimilar device shadowing (DDS). To use DDS, all systems that have mounted a shadow set whose members differ in size must be running OpenVMS Alpha Version 7.3-2.

Prior to OpenVMS Alpha Version 7.3-2, Volume Shadowing for OpenVMS required that all members of a shadow set be the same size, that is, that each member have the exact same number of blocks. The rapid advance of disk technology has made this requirement impractical. The flexibility of using different size devices outweighs the space that will be unused on the larger device.

Operationally, shadowing dissimilar devices means that you can add a larger disk device to an existing shadow set. The shadow set retains the file system size of the original shadow set. After adding a larger disk, if you remove a smaller disk, the geometry (cylinders, tracks, and sections) of the shadow set changes to the smallest remaining disk, but the logical volume size (that is, the file system size) is not changed.

A complementary new feature, dynamic volume expansion (DVE), enables you to increase the logical volume size (amount of disk space allocated to the file system) without dismounting the device.

The basis of dynamic volume expansion is the one-time allocation of extra bitmap space to the maximum size that will ever be used on this volume (that is, 1 TB, if you want). This can be performed at disk initialization time or by using the SET VOLUME command. This makes it possible to expand the volume size at a later time while the device is mounted. Note that to use the SET VOLUME command to increase the expansion limit, the disk must be mounted privately. However, once the expansion limit is set, the volume can be expanded while the disk is mounted as shareable (MOUNT/SHARE).

Together, the features of dissimilar device shadowing and dynamic volume expansion enable you to continually grow a logical volume without ever having to take it off line again.

For more information about shadowing dissimilar devices, refer to *HP Volume Shadowing for OpenVMS*. For information about dynamic volume expansion, refer to the *HP OpenVMS System Manager's Manual*.

Error Log Viewer (ELV) Utility

The new Error Log Viewer (ELV) utility allows you to quickly examine, from the command line, an error log file in a user-readable format before deciding whether the data warrants a more comprehensive analysis with a tool such as the System Event Analyzer.

ELV is particularly useful with error logs created on systems with newer devices or on newer systems such as most of the AlphaServer DS, ES, and GS series. ELV can also be used with error logs created on some older systems or on systems equipped with older devices.

ELV provides detailed information for all error log events of the following types:

- Bugchecks
- Volume mounts and dismounts
- Correctable error throttling notifications
- Software parameters
- Control entries
  - — Indictment events
  - — System service messages
  - — Network messages
  - — Operator messages
  - — ERRLOG.SYS created
  - — Time stamps
  - — System startup messages

ELV provides detailed information for some error log events of the following types:

- Machine checks
- Device errors
- Device timeouts
- Asynchronous device attentions
- Logged messages
- Logged MSCP messages

For more information, refer to the *HP OpenVMS System Management Utilities Reference Manual* and to the online help for ELV.

Euro Symbol and Additional Font Sets Support

OpenVMS Alpha Version 7.3-2 offers support for additional fonts with the installation of the DECwindows Server component. The following fonts are now available along with the installation of the core set of 100 dpi, 75 dpi, and common font sets:

- Agfa Monotype Windows-compatible TrueType fonts

  To ensure fast, high-quality text-rendering capabilities, OpenVMS Alpha Version 7.3-2 includes the iType font rendering technology from Agfa Monotype Corporation along with a number of scalable fonts, including the Albany, Cumberland, Screen, and Thorndale type families. These fonts are identical in screen and printer metrics to the Windows core fonts Arial, Courier, and Times New Roman. Agfa Monotype Windows-compatible fonts are part of the Enhanced Screen Quality (ESQ) line of TrueType fonts optimized for viewing at any resolution.

  For more information about the iType technology, see the Agfa Monotype web site:

  `http://www.agfamonotype.com`

- X11R6.6 fonts

  As part of the standard X11R6.6 implementation, OpenVMS Alpha Version 7.3-2 includes the 75- and 100-dpi versions of the Bitstream Charter and Adobe® Utopia font families.

- Previously undocumented fonts

  These fonts include the 75-dpi, 100-dpi, and common versions of the Lucida, Present Bullets, Fixed Width, Sun Open Look Glyph, and VT330 font families, as well as a set of language-specific and miscellaneous fonts. Also included are the Bitstream Speedo, Adobe Type1 Courier, and the Utopia Scalable fonts.

Font sets that support the euro currency symbol, which were formerly provided through a separate remedial kit, are also available during installation of the DECwindows Server component. You can either choose to install the fonts as part of the default installation or choose the Euro Base Support option during a custom installation. Note that once the fonts are installed, you must enable the fonts to make them available for use.

See the *HP DECwindows Motif for OpenVMS Alpha New Features* manual for instructions on enabling the euro font sets, as well as the file location and X logical Font Descriptions (XLFDS) for the new fixed-width and scalable fonts.

Extended File Cache (XFC) Enhancements

Enhancements to XFC in OpenVMS Version 7.3-2 fall into these categories:

- Performance improvements
- Usability improvements

Specific enhancements are discussed in the following sections.

XFC Performance Enhancements

Three XFC enhancements have contributed to significant increases in speed in this version of the OpenVMS operating system:

- Faster freeing of memory

  When a file is deleted or a volume containing a file is dismounted, XFC releases the memory used for cache data on the current node. This operation, called deposing cache data, has been speeded up dramatically in this version of OpenVMS — in some cases, by a factor of 10.

- Deposing volumes speeded up

  In addition to speeding up the deposing of cache data, the first file was formerly deposed asynchronously, which at times caused noticeable dismount delays. This synchronization problem has been corrected.

- Improved performance of noncached I/Os

  In previous releases, XFC added overhead for I/Os that were not cached — for example, very large I/Os of 6000 blocks. This extra overhead has been removed.

XFC Usability Improvements

Two types of enhancements have contributed to improvements in the usability of XFC:

- Size of counters increased

  The XFC public counter that the DCL command SHOW MEMORY/CACHE uses was stored in unsigned longwords, thus limiting the maximum counts to about 4 billion. These counters have been increased to unsigned quadwords. Users now see a display of up to 8 bytes of data for each counter.

- XFC SDA display enhancements

  These enhancements include the following:

  — Help for the XFC SDA extension has been updated.

  — The SDA command XFC SHOW FILE now displays the file name.

  — The output of the SDA command XFC SHOW FILE/BRIEF is sorted by volume.

Fast Path for LAN Devices

With OpenVMS Version 7.3-2, further enhancements have been made to Fast Path for LAN devices, which will continue to help streamline I/O processing and improve symmetric-multiprocessing (SMP) performance scalability on newer AlphaServer systems. Enhancements include:

- Reduced contention for the SCS/IOLOCK8 spinlock. The LAN drivers now synchronize using a LAN port-specific spinlock where possible.
- Offload of the primary CPU. The LAN drivers may be assigned to a secondary CPU so that I/O processing can be initiated and completed on the secondary CPU. This offloads the primary CPU and reduces cache contention between processors.

These features enhance the Fast Path functionality that already exists in LAN drivers. The enhanced functionality includes additional optimizations, preallocating of resources, and providing an optimized code path for mainline code.

Fast Path Support for Smart Array 53*xx* Series Adapter (KZPDC)

OpenVMS Alpha Version 7.3-2 supports KZPDC, a Smart Array 53*xx* series backplane RAID controller that uses Ultra2 SCSI 160 MB/s disks. The KZPDC is supported for direct-attached storage only. PKRdriver is the software component that supports KZPDC. OpenVMS Alpha Version 7.3-2 enhancements provide Fast Path support for the KZPDC. Fast Path support is designed for use in a multiprocessor system and automatically redirects I/O from the primary CPU to different CPUs. You can change it either programmatically or by using DCL commands.

For more information, refer to the *HP OpenVMS System Manager's Manual.*

Fibre Channel New Features

The following new features for Fibre Channel configurations are introduced in this release.

MSA1000 Support

OpenVMS Alpha Version 7.3-2 supports the HP StorageWorks modular SAN array 1000 (MSA1000). The MSA1000 features include:

- 2-GB Fibre Channel front end
- 4U 160 SCSI backend ports
- 4U rackmount with 14 drives
- 28 additional drives with 2 external storage shelves
- Compatible with existing SANs

The MSA1000 is supported on OpenVMS Alpha Versions 7.2-2, 7.3, and 7.3-1 with the latest FIBRE_SCSI patch kit. In the future, an arbitrated loop configuration will be supported, as described in the *Guidelines for OpenVMS Cluster Configurations* manual.

For more information about the HP StorageWorks modular SAN array 1000 for AlphaServer systems, see the QuickSpec at:

`http://www.hp.com/country/us/eng/prodserv/storage.html`

New FIBRE_SCAN Utility

A new standalone program called FIBRE_SCAN.EXE is included in OpenVMS Alpha Version 7.3-2. The purpose of this program is to display information about all storage devices attached to Fibre Channel on the system; both configured and nonconfigured devices are included.

The program can be invoked in two modes:

```
$ MCR SYS$ETC:FIBRE_SCAN          !Scans all ports on the Fibre Channel
$ MCR SYS$ETC:FIBRE_SCAN PGx      !Scans only port x on the Fibre Channel
```

FIBRE_SCAN requires CMKRNL and LOG_IO privilege. To capture the output in a file, the user can first enter the program with a command such as the following:

```
$ define/user sys$output xxx.log
```

Note also that FIBRE_SCAN is a display-only utility and cannot be used to load device drivers or to configure devices on the Fibre Channel.

FIBRE_SCAN can be used locally on each system. It cannot be used on systems running versions prior to OpenVMS Alpha Version 7.3-2, and it cannot be used to display devices attached to other systems in a cluster.

Gigabit Ethernet Adapter (DEGXA) Support

OpenVMS Alpha Version 7.3-2 supports the DEGXA Gigabit Ethernet adapter, which is a Broadcom BCM5703 chip (TIGON3) network interface card (NIC). The introduction of the DEGXA Gigabit Ethernet adapter continues the existing Gigabit Ethernet support as both a LAN device as well as a cluster interconnect device.

Simplified Installation, Upgrade, and Configuration Procedures

The installation, upgrade, and configuration procedures have been simplified to include fewer options. Approximately 25 options have been eliminated. The *HP OpenVMS Alpha Version 7.3-2 Upgrade and Installation Manual* includes the current list of options. The result is a smoother, more streamlined procedure. If you choose to list options or to change the selected options, you have fewer options to review.

CDSA and Kerberos in OpenVMS Upgrades and Installations

CDSA and Kerberos are provided in separate kits but are required parts of the OpenVMS operating system. During installation or upgrade, CDSA and Kerberos are listed as separate products. Do not attempt to remove them from the options list.

Enhanced Time Zone Selection Procedure (UTC$TIME_SETUP.COM)

SYS$MANAGER:UTC$TIME_SETUP.COM has been modified and enhanced. This procedure is used during an OpenVMS installation and possibly when upgrading the OpenVMS system. The procedure can be invoked independently from a running system.

The menu displays have been modified so that more choices can fit in a single menu display. In addition, a search option has been added that allows you to bypass the menus and search for a time zone by name.

To invoke the search option, enter an equals sign (=), followed by the full or partial name of the time zone, for example, =EASTERN. If you enter just the equals sign (=), you are prompted to enter a full or partial time-zone name. A list of matching time zones then appears from which you can select the desired time zone. Example 3-2 illustrates the use of this new feature.

Example 3-2 Selecting a Time Zone

```
Search the Time Zone by Full or Partial Name

"*" indicates the menu
```

```
                        1)  EASTERN / CANADA
                        2)  EASTERN / US

          Press "Return" to redisplay this menu,
          enter "=" to search for a new zone,
          enter "0" to return to the Main Time Zone Menu,
          enter "?" for help, or
          Select the number above that best represents the desired time zone: 2

          You selected EASTERN / US as your time zone.
           Is this correct? (Yes/No)  [YES]:

Select the number above that best represents the desired time zone: =eastern
```

LAN Failover

LAN Failover in OpenVMS Alpha Version 7.3-2 provides a mechanism to protect your system against a network interface card (NIC) failure. This mechanism provides a failover safeguard for all LAN applications by integrating individual network adapters on the same local network into a single virtual interface called a LAN Failover set.

The system manager defines and creates the LAN Failover set. The failover set consists of one adapter that is to be used for LAN traffic and one or more adapters that remain idle. If the active adapter fails, one of the idle set members automatically takes over with the same address, thereby allowing continuous operation.

LAN Failover determines the status of network interfaces through a virtual driver. It determines which network adapters are functioning and selects one to maintain the LAN traffic. All network protocols can benefit from this type of failover.

LAN Failover enhances the existing disaster-tolerance capabilities of OpenVMS by providing users with a high-availability solution for all LAN applications. For more information about LAN Failover, refer to the *HP OpenVMS System Manager's Manual*, the *HP OpenVMS System Management Utilities Reference Manual*, and the *HP OpenVMS I/O User's Reference Manual*.

failSAFE IP

The failSAFE IP Version 5.4 feature is another solution for network connectivity failures available to OpenVMS Version 7.3-2 customers running TCP/IP.

failSAFE IP, which ships with HP TCP/IP Services for OpenVMS Version 5.4 product, migrates IP addresses from a failed network interface card (NIC) to a preconfigured NIC on the same node or another node in the cluster. Multiple NICs can be configured with IP addresses in the same subnet. This provides an additional benefit by load-balancing outbound TCP connections and by providing higher aggregate throughput. failSAFE IP can be used only in a TCP/IP environment. For more information about failSAFE IP, refer to the TCP/IP Services for OpenVMS documentation.

Table 3-3 compares the features of LAN Failover and failSAFE IP.

Table 3-3 LAN Failover and failSAFE IP Comparison

Feature	LAN Failover	failSAFE IP
NIC usage	One NIC active, others are standby.	All NICs active, load balancing.
Devices supported	Ethernet device types – DEGXA, DEGPA, and DE600 series.	Independent of device types.
Protocols	Supports all LAN protocols.	Available only on TCP/IP.
Failover time	Typically milliseconds.	Typically a few seconds.

New Mail Flag

OpenVMS Alpha Version 7.3-2 includes a new Mail utility flag, RETRY_UAF, which allows you to control whether a record-locked read failure in MAIL on a UAF record is retried indefinitely or not. Setting this flag prevents %MAIL-E-UAFGETERR errors from occurring due to locked UAF records in Mail processing environments that cannot tolerate this type of error.

To set the RETRY_UAF flag (and keep the settings of the existing flags):

1. Add 32 to the current value of the MAIL$SYSTEM_FLAGS system logical
2. Redefine the MAIL$SYSTEM_FLAGS system logical to the new value.

The default behavior (bit not set) returns UAFGETERR and does not perform a retry.

Mount Verification Message Reduction

In a Storage Area Network (SAN), mount verification takes place for a variety of reasons, including:

- Path switch by another cluster node
- Dropped Fibre Channel packets (an infrequent occurrence)
- Rezone of a SAN, which causes in-flight I/O to be dropped

Quieter mount verification suppresses the messages that previously were displayed for mount verification events from which the devices immediately recovered. These messages alarmed some customers.

The number of messages logged to the operator's log is now controlled by two system parameters: MVSUPMSG_NUM and MVSUPMSG_INTVL. MVSUPMSG_NUM specifies a number of mount verifications; MVSUPMSG_INTVL specifies a duration in seconds. If the number of mount verifications for a given device meets or exceeds the number specified by MVSUPMSG_NUM within the time specified for MVSUPMSG_INTVL, then an OPCOM message is issued.

Customers who prefer the prior behavior or who would like to increase or decrease the number of messages that are logged can adjust the system parameter settings.

For more information about these new system parameters, refer to the *HP OpenVMS System Management Utilities Reference Manual*.

PEdriver Fast Path Support

PEdriver, the software that enables OpenVMS Cluster communications over a LAN, now provides Fast Path support. This new PEdriver feature provides the following benefits:

- Improves SMP performance scalability.
- Reduces the contention for the SCS/IOLOCK8 spinlock. PEdriver uses a private port mainline spinlock to synchronize its internal operation.
- Allows PEdriver to perform cluster communications processing on a secondary CPU, thus offloading the primary CPU.
- Allows PEdriver to process cluster communications using a single CPU.
- Reduces CPU cost by providing a Fast Path streamlined code path for DSA and served blocked data operations.

For more detailed information, refer to the *HP OpenVMS I/O User's Reference Manual*, the *HP OpenVMS System Manager's Manual*, and the *HP OpenVMS System Management Utilities Reference Manual*.

POLYCENTER Software Installation New Features

The following sections describe enhancements to the PRODUCT command for the POLYCENTER Software Installation utility provided with the OpenVMS Alpha Version 7.3-2 operating system.

Running in Recovery Mode

You can use the new /RECOVERY_MODE qualifier with the PRODUCT INSTALL and PRODUCT RECONFIGURE commands to recover automatically from an abnormal termination of either operation. If the operation fails or if you terminate the operation with Ctrl/C or Ctrl/Y, the utility tries to restore your system environment to its original state. The utility attempts to reinstate files, directories, and libraries that were created, modified, or deleted prior to the failure.

Uninstalling a Patch

The POLYCENTER Software Installation utility contains the following new commands:

- PRODUCT DELETE RECOVERY_DATA — Deletes one or more patch recovery data sets.
- PRODUCT SHOW RECOVERY_DATA — Displays patch recovery data sets in chronological order, newest to oldest.
- PRODUCT UNDO PATCH — Uninstalls one or more patches for which recovery data has been saved.

Using the PRODUCT INSTALL /SAVE_RECOVERY_DATA command to install patch and mandatory update kits creates a recovery data set that can be used to *undo* the operation. Restrictions apply to the use of recovery data sets. When used to uninstall (or roll back) patches, recovery data sets must be processed in the reverse order from which they were created. Also, recovery data sets become invalid whenever a PRODUCT command modifies the product database, except when you use the PRODUCT INSTALL /SAVE_RECOVERY_DATA command to install additional patches or mandatory updates.

The following command installs an OpenVMS remedial kit and creates a recovery data set for it:

```
$ PRODUCT INSTALL VMS732_UPDATE /SAVE_RECOVERY_DATA
```

In this example, the /SAVE_RECOVERY_DATA qualifier causes all the directories, files, and modules that are deleted or modified in the process of installation to be saved in a directory tree named [PCSI$UNDO_001] on the system disk. A copy of the product database is also saved in this directory tree, and its contents constitute a recovery data set.

Recovery data sets can be used at a later time to uninstall patches and mandatory updates by using the PRODUCT UNDO PATCH command. The newest recovery data set is always number 001, the next oldest is renamed to 002, and so on.

For more detailed information, refer to the command descriptions in the *HP OpenVMS System Management Utilities Reference Manual* or to the online help for the PRODUCT command.

Data Compression

The POLYCENTER Software Installation utility supports product kits in three formats:

- Compressed format

  A data compression technique has been applied to a sequential kit. A compressed kit has a .PCSI$COMPRESSED file type.

- Reference format

  Product files exist in a directory tree. A .PCSI$DESCRIPTION file in the top level of the directory tree denotes a reference kit.

- Sequential format

  Product files are placed in a container file. A file type of .PCSI indicates a sequential kit.

You use the PRODUCT COPY command to create a product kit in compressed format from an existing kit in sequential format. For example:

```
$ PRODUCT COPY WIDGET /SOURCE=[dir1] /DESTINATION=[dir2] /FORMAT=COMPRESSED
```

You can compress kits that were created by an earlier version of the utility.

The processing of compressed kits is transparent to the user. During installation, records from the kit are decompressed (that is, expanded) automatically as they are read by the utility. No separate steps, intermediate storage, or new qualifiers are required to perform typical operations on compressed kits.

If kits in both sequential and compressed format for the same product are placed in the source directory, PRODUCT commands use the kit in compressed format by default. You can use the /KIT_ATTRIBUTES=FORMAT=*keyword* qualifier with many PRODUCT commands to choose kits of a particular format.

Display of File and Module Generation Numbers

When two or more products (or two or more patches to the same product) supply the same file or library module, the POLYCENTER Software Installation utility uses the generation attribute specified in the product kit or product database for the object to resolve any conflicts.

The object with the highest generation number supersedes the others. If the numbers are nonzero and a tie exists, the objects are considered to be identical. If one of the objects in conflict has a generation number greater than zero, then the conflict cannot be resolved.

Prior to OpenVMS Alpha Version 7.3-2, the only way for a user to find out the generation information for a file or module was to extract and examine the product description file (PDF) from the kit. Now this information is displayed by default for the PRODUCT SHOW OBJECT command.

SET DEVICE Command Can Reset Device Counts

User with CMKRNL and DIAGNOSE privileges can use a new /RESET qualifier with the SET DEVICE command to reset either or both the error count and the operation count for a device. Use the following format:

SET DEVICE [*device-name*[:]] /RESET=(*keyword*[, *keyword*])

You must specify one or both of the following keywords:

- ERROR_COUNT — Resets the error count.
- OPERATION_COUNT — Resets the operation count.

When either count is reset, a message is sent to the error log. The message indicates that the count was reset and includes the value of the counter before the reset.

For more information about the SET DEVICE command, refer to online help or to the *HP OpenVMS DCL Dictionary*.

SYSMAN DUMP_PRIORITY Commands

The new SYSMAN DUMP_PRIORITY commands allow system managers and developers to register with BUGCHECK the names and UICs of their key processes. If a bugcheck then occurs, BUGCHECK uses this list of priority processes to make sure these processes are dumped as early as possible in a selective dump. (They will be dumped immediately following any current processes and the swapper.) This increases the probability that these processes will fit into a selective dump.

For more information, refer to the SYSMAN utility documentation in the *HP OpenVMS System Management Utilities Reference Manual* and to the online help for SYSMAN.

System Parameters

This section describes changes to the system parameters, BALSETCNT and WSMAX, and introduces two new system parameters, SHADOW_REC_DLY and SHADOW_SITE_ID.

BALSETCNT and WSMAX System Parameters

Prior to OpenVMS Alpha Version 7.3-2, it was necessary for large systems to make tradeoffs between having a large balance set count (BALSETCNT) or large working sets (WSMAX). The need for making these tradeoffs has been removed in OpenVMS Alpha Version 7.3-2. You can now have systems with both very large balance set counts and very large working sets.

In addition, the BALSETCNT system parameter has now become a dynamic parameter. This parameter is no longer a strict setting of the number of processes that may be resident in memory. The swapper will try and reduce the number of resident processes down to BALSETCNT. However, if the total number of active processes and processes that have disabled swapping exceeds BALSETCNT, the swapper will not force processes out of memory just to meet the BALSETCNT setting.

SHADOW_REC_DLY System Parameter

The new system parameter SHADOW_REC_DLY represents the number of seconds a system waits before it attempts to manage transient state operations on any virtual units that are mounted on this system. A shadow set enters a transient state when a merge or a copy operation is required on that virtual unit.

SHADOW_SITE_ID System Parameter

System managers can now define the site value to be used for all shadow sets mounted on a system. This parameter is an arbitrary numeric value coordinated by the system manager of disaster-tolerant clusters. Reads from devices that have site values matching the shadow set's site value are preferred over reads from devices with different site values.

For more information, refer to the description of the SET DEVICE/SITE command in the *HP OpenVMS DCL Dictionary* and to the *HP Volume Shadowing for OpenVMS* manual.

UNIX Portability Security Features

The following new features have been added to the UNIX portability effort:

- Support for GUID and UID UNIX security identifiers
- Support for mixed-case passwords
- New screen lock and idle process killers shipped with CDE

USB Configuration Manager (UCM) Utility

The new USB Configuration Manager (UCM) utility is an OpenVMS utility for EV7 AlphaServer systems. UCM does the following:

- Records events such as plugging or unplugging devices and errors that occur on a USB bus. This is the USB event-logging function of UCM.
- Maps physical devices to persistent device names (based on either serial number or bus location).
- Manages additions, deletions, and modifications to devices configured on the system.

For more information, refer to the *HP OpenVMS System Management Utilities Reference Manual* and to the online help for UCM.

Volume Shadowing Management Functions

Many new functions have been created for managing shadow sets created with Volume Shadowing for OpenVMS. These functions are implemented with two new DCL commands: SET SHADOW and SHOW SHADOW. Together, these functions are known as the SETSHOSHADOW utility. In addition, the ANALYZE/DISK utility has been extended for analyzing shadow sets.

Customers have requested greater control over the order of copy and merge operations. They also want to be able to specify on which nodes copy and merge operations take place. SET SHADOW, with its many qualifiers, gives customers these capabilities. SHOW SHADOW displays a variety of information about a shadow set, including the merge or copy status of the specified shadow set.

ANALYZE/DISK/SHADOW can be used to examine a specified range of blocks in a shadow set or the entire contents of a shadow set.

For more information, refer to the following manuals:

- *HP Volume Shadowing for OpenVMS*

  Describes the new commands and new qualifiers, along with information about how, when, and why to use them.

- *HP OpenVMS DCL Dictionary*

  Describes the SET SHADOW and SHOW SHADOW commands and qualifiers.

- *HP OpenVMS System Management Utilities Reference Manual*

  Describes the ANALYZE/DISK utility's new qualifier, /SHADOW, and its keywords.

4 Programming Features

This chapter describes the following new features related to application and system programming on this version of the OpenVMS operating system:

- Authentication and Credentials Management Extensions (ACME) subsystem SDK
- C Run-Time library enhancements
- DDT Intercept Establisher Routines
- DECdtm/XA Version 2.1
- $GETDVI item code DVI$SHDW_SITE
- $GETRMI system service calls
- Fast Path for LAN devices
- Large pool lookaside lists
- POSIX Threads NUMA support
- PEdriver Fast Path support
- System Dump Analyzer (SDA) commands

Authentication and Credentials Management Extensions (ACME) Agent Software Development Kit (SDK)

The Authentication and Credentials Management Extensions (ACME) subsystem provides authentication and persona-based credential services. Applications use these services to enforce authentication policies defined by ACME agents running in the context of the ACME_SERVER process.

There are two ACME agents available today: VMS (standard OpenVMS policy) and MSV1_0 (Microsoft LAN Manager authentication). In the future, customers will be able to create additional ACME agents for custom authentication policies.

A field test version of the ACME agent software development kit (SDK) is available for customers who would like to evaluate the ACME agent interface and programming environment as it is implemented in this version of OpenVMS.

NOTE The ACME agent interface and programming environment is subject to change until finalized in a future OpenVMS release. User-defined ACME agents will not be supported in a production environment until that time.

An evaluation version of the LDAP ACME agent is included in the ACME Agent SDK. The LDAP ACME agent allows users to log into an OpenVMS system using authentication information held in an LDAP directory, thus allowing common authentication across many OpenVMS systems.

NOTE The LDAP ACME agent is for evaluation purposes only.

For more information about the ACME Agent SDK, refer to the SYS$HELP:ACME_DEV_README.TXT file.

C Run-Time Library Enhancements

The following sections describe the C Run-Time Library (CRTL) enhancements included in OpenVMS Alpha Version 7.3-2. These enhancements provide improved UNIX portability and the flexibility of additional user-controlled feature selections. New C RTL functions are also included.

POSIX Style Identifiers

POSIX style identifier support is added to the C RTL, providing additional UNIX portability and standards compliance. POSIX style identifiers refer to the user identifier (UID), group identifier (GID), and process group. The scope includes real and effective identifiers.

The support for POSIX style identifiers in the C RTL requires 32-bit user and group ID support and also depends on features in the base version of OpenVMS. POSIX style IDs are supported by OpenVMS Alpha Version 7.3-2.

To use POSIX style identifiers, applications must be compiled for 32-bit UID/GID (by defining the __USE_LONG_GID_T macro), and POSIX style IDs must be enabled by defining the DECC$POSIX_STYLE_UID feature logical name to ENABLE, as shown in the following command:

```
$ DEFINE DECC$POSIX_STYLE_UID ENABLE
```

To disable POSIX style IDs, define the DECC$POSIX_STYLE_UID logical to DISABLE:

```
$ DEFINE DECC$POSIX_STYLE_UID DISABLE
```

Socket Enhancements

The following socket enhancements are included:

- 64-bit pointer support is added for the following socket routines:

  ```
  freeaddrinfo      recvmsg
  getaddrinfo       sendmsg
  ```

  Previously, these functions had only a 32-bit interface.

Set Default Directory for Child Processes

You can now set the default (working) directory for child processes created by `vfork/exec*`, using the new C RTL function `decc$set_child_default_dir`. Ordinarily, in the absence of this function call, child processes inherit the default directory of their parent process. By calling `decc$setchild_default_dir` (default_dir), all child processes created with `vfork/exec*` have their default directory set to `default_dir` when they begin execution. The `default_dir function` must represent a valid directory specification, or results of the call are unpredictable (the child process might fail without notification). Both OpenVMS and UNIX style file specifications are supported for this function call.

You can reset the inherited working directory to that of the parent by executing `decc$set_child_default_dir` (NULL). Subsequently, all created child processes will inherit their parent's working directory. This is the normal default behavior for `vfork/exec*`.

New Logicals

The following new logicals provide additional standards conformance and UNIX portability:

- DECC$UNIX_LEVEL provides a mechanism to specify a level of UNIX compatibility that allows an application to use a new feature provided for UNIX compatibility without having to set each of the individual C RTL UNIX compatibility feature logicals.
- DECC$ENABLE_TO_VMS_LOGNAME_CACHE speeds the translation of logical names in UNIX name translation.
- DECC$EFS_NO_DOTS_IN_DIRNAME suppresses the interpretation of file names that contain dots as directory names.
- DECC$ALLOW_REMOVE_OPEN_FILES controls the behavior of the `remove` function on open files. Ordinarily the operation fails. However, POSIX conformance dictates that the operation succeed. This POSIX conformant behavior is enabled by the new feature switch.
- DECC$POSIX_STYLE_UID controls whether 32-bit UIDs and GIDs are interpreted as POSIX style identifiers or derived from the process UIC.
- DECC$NO_ROOTED_SEARCH_LISTS controls how the `to_vms` function resolves search-list logicals.
- DECC$ALLOW_UNPRIVILEGED_NICE controls whether the `nice` function exhibits its legacy behavior of not including a privilege check on the calling process, or whether it behaves according to the X/Open specification by checking the privilege.
- DECCUSE_JPI_CREATOR determines the parent process ID in `getppid` by calling $GETJPI using item JPI$CREATOR instead of JPI$OWNER. This feature is available only on systems that support POSIX style session identifiers.
- DECC$WRITE_SHORT_RECORDS accommodates short-sized records while retaining the legacy way of writing records to a fixed-length file as the default behavior.
- DECC$PIPE_BUFFER_QUOTA lets you specify a buffer quota for the `pipe` function to use if the new, optional fourth parameter of that function is omitted.
- DECC$GLOB_UNIX_STYLE selects the UNIX mode of the `glob` function.
- DECC$ACL_ACCESS_CHECK feature logical affects whether or not the `access` function checks for OpenVMS access control lists (ACLs) as well as UIC protection.
- DECC$RENAME_ALLOW_DIR feature logical affects the behavior of the `rename` function.
- DECC$EXEC_FILATTR_INHERITANCE feature logical is modified to allow greater choice in setting process inheritance for file access modes.

Performance Enhancements

Various enhancements to improve performance, especially through faster UNIX name translation, are included:

- A logical name cache is implemented to reduce the high cost of translating logical names in UNIX name translation. This cache is used only when the new DECC$ENABLE_TO_VMS_LOGNAME_CACHE feature logical is enabled.

- A feature is added to the C RTL to suppress interpretation of file specifications that contain dots as directory names. This feature is activated by enabling the new DECC$EFS_NO_DOTS_IN_DIRNAME feature logical.
- The `decc$translate_vms` function has been optimized for faster translation of file names.

Additional Enhancements

The following additional functional enhancements are included:

- The C RTL can now handle command lines up to 4K bytes (up from 256 bytes).
- Support is added for angle brackets in equivalence strings for logical names used in UNIX name translation.

New and Changed Functions

The following new C RTL functions are provided with OpenVMS Alpha Version 7.3-2:

File Read/Write Functions:

```
pread            readv            _writev64
pwrite          _readv64
```

Print Functions:

```
snprintf
vsnprintf
```

Signal Functions:

```
sighold       sigignore        sigrelse
sigwait       sigtimedwait     sigwaitinfo
```

Time Functions:

```
nanosleep     clock_gettime
clock_getres  clock_settime
```

Password Functions:

```
getpwnam_r    _getpwnam_r64   __64_getpwnam  __64_getpwent
getpwuid      getpwuid_r      __64_getpwuid   _getpwuid_r64
```

Security/Impersonation Functions (from COE):

```
endgrent      getgrnam        getsid        setpgrp
getgrent      getgrnam_r      seteuid       setregid
getgrgid      getpgid         setgrent      setreuid
getgrgid_r    getpgrp         setpgid       setsid
```

Socket Function:

```
poll
```

Miscellaneous Functions:

```
a64l          ttyname_r       glob              globfree
l64a          rand_r          decc$set_child_default_dir
```

The following modified C RTL functions are provided with OpenVMS Alpha Version 7.3-2:

```
endpwent      setgid          geteuid
getegid       setpwent        getgid
getpwent      setuid          getuid
ttyname_r
```

From more information, refer to the *HP C Run-Time Library Reference Manual for OpenVMS Systems*.

DDT Intercept Establisher Routines and Device Configuration Notification Routines

OpenVMS Alpha Version 7.3-2 introduces new routines for use in OpenVMS applications that are developed by third-party application providers. These new routines are designed for establishing driver dispatch table (DDT) intercepts of OpenVMS device drivers and for providing notification of device configuration by way of a callback. These routines can be used by any privileged kernel-mode application that alters the DDT. The routines provide a new method for intercepting calls into the driver by way of DDT entry points so that multiple intercepts work correctly. Note that the routines for providing notification of device configuration (and for revoking notification) are not limited to use with the DDT intercept establisher routines.

These routines, when used in third-party kernel-mode applications such as disk caching products and SCSI disk-shadowing applications, enable these applications to run in an OpenVMS SCSI or Fibre Channel multipath configuration. Any third-party applications that rely on altering the DDT of the OpenVMS Alpha SCSI disk-class driver (SYS$DKDRIVER.EXE), the SCSI tape-class driver (SYS$MKDRIVER), or the SCSI generic-class driver (SYS$GKDRIVER) will require source changes to use these new routines.

If you are using a third-party disk-caching product or disk-shadowing application, avoid using it in an OpenVMS SCSI or Fibre Channel multipath configuration until you confirm that the application has been revised to use these new routines.

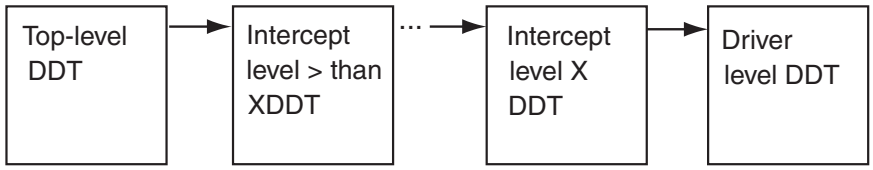
These new routines provide a solution to a problem that developed after the introduction of SCSI and Fibre Channel multipath support in OpenVMS Alpha Version 7.2-1. Customer were advised to avoid using these third-party SCSI disk-caching and disk-shadowing applications when their SCSI devices were configured for multipath failover. The new routines enable providers of third-party application providers to modify their applications so that they function correctly.

DDT Intercept Establisher Routines

New routines establish the DDT intercepts. Third-party applications that modified the DDT directly can be modified to use the appropriate establisher routines, so that they function properly in an OpenVMS SCSI or Fibre Channel configuration. These routines establish intercepts on a per-UCB basis.

At a given entry point, there can be multiple declarations of DDT intercepts. The following figure illustrates multiple DDT declarations.

Figure 4-1 DDT Intercepts



The new DDT intercept establisher routines are:

- IOC_STD$ESTABLISH_DDT_CANCEL
- IOC_STD$ESTABLISH_DDT_ALTSTART
- IOC_STD$ESTABLISH_DDT_START
- IOC_STD$ESTABLISH_DDT_MNTVER

If there are multiple declarations of DDT intercepts, they are called in descending order, from the highest level DDT (DDT$K_ITCLVL_TOP) to the lowest-level DDT (DDT$K_ITCLVL_DRVR).

Although the standard driver `cancel`, `altstart`, `start`, and `mntver` routines do not return a status, the intermediate routines must return either SS$_SUPERSEDE or SS$_CHAINW status. Any other return value results in a bugcheck. As the return value suggests, the SS$_SUPERSEDE return value from the intermediate routine supersedes the lower-level call to the DDT intercept routines. The SS$_CHAINW return value from the intermediate routine causes the next lower-level DDT intercept routines to be called.

The intercept DDTs are placed in the DDT chain according to their level. The top-level DDT is always the dispatcher DDT, and the bottom-level DDT is always the driver-level DDT. Other DDTs are placed in descending order between the top-level DDT and the driver-level DDT.

Intercept Levels

The following intercept levels are currently defined and reserved for use by HP:

```
DDT$K_ITCLVL_TOP      32767
DDT$K_ITCLVL_HSM      24576
DDT$K_ITCLVL_MPDEV    4096
DDT$K_ITCLVL_DRVR      0
```

The valid intercept levels are from 4097 to 32766, except for 24576, which is reserved for the HSM interval. You can define as many intercepts as needed in that range.

Restrictions

The following restrictions exist:

- Third-party intercepts are allowed only in the primary path UCB.
- Multipath currently does not support intercepts in the secondary path UCB.
- The flag parameter is required. It is a placeholder for future development, and the value must be zero.

Manufacturers of third-party products that rely on altering the DDT of either the OpenVMS Alpha SCSI disk-class driver (SYS$DKDRIVER.EXE), the SCSI tape-class driver (SYS$MKDRIVER), or the SCSI generic-class driver (SYS$GKDRIVER) can contact HP at `vms_drivers@zko.dec.com` for more information.

Routines

The following pages provide the functional description of each routine.

IOC_STD$ESTABLISH_DDT_START

Establishes DDT$PS_START_2 intercept.

Functional Description

This routine establishes the intercept of the DDT$PS_START_2 routine.

Calling Convention

```
int  ioc_std$establish_ddt_start (UCB *ucb, int (*start_itc_routine)(IRP *irp,
     UCB *ucb), int level,int flag)
```

Input

ucb	Pointer to a UCB whose DDT$PS_START_2 is to be intercepted.
start_itc_routine	The intercepting `start` routine. This routine is called before the driver's `start` routine. The calling convention of the `start` routine is the same as the standard DDT$PS_START_2 routine, except that this routine must return one of the following status values: SS$_CHAINW - The next `start` routine should be called. SS$_SUPERSEDE - No more `start` routines should be called. Any other return value results in a bugcheck.
level	Level of DDT to be intercepted. Currently, multipath does not support an intercept level below MPDEV intercept.
flag	Placeholder for future development; must be zero.

Return Value

SS$NORMAL	DDT intercept added successfully. This routine may also return various other error status values, including any status returned on a failure to allocate pool.

Synchronization Environment

Caller must be in kernel mode, IPL at or below UCB fork IPL.

Almost all use of the DDT within OpenVMS requires holding the UCB fork lock. This is why this routine acquires and conditionally releases the UCB fork lock to change the DDT.

IOC_STD$ESTABLISH_DDT_ALTSTART

Establishes DDT$PS_ALTSTART_2 intercept.

Functional Description

This routine establishes the intercept of the DDT$PS_ALTSTART_2 routine.

Calling Convention

```
int  ioc_std$establish_ddt_altstart (UCB *ucb, int (*altstart_itc_routine)(IRP *irp,
     UCB *ucb), int level,int flag)
```

Input

ucb	Pointer to a UCB whose DDT$PS_ALTSTART_2 is to be intercepted.
altstart_itc_routine	The intercepting `altstart` routine. This routine is called before the driver's `altstart` routine. The calling convention of the `altstart` routine is the same as the standard DDT$PS_ALTSTART_2 routine, except that this routine must return one of the following status values: SS$_CHAINW - The next `altstart` routine should be called. SS$_SUPERSEDE - No more `altstart` routines should be called. Any other return value results in a bugcheck.
level	Level of DDT to be intercepted. Currently, multipath does not support an intercept level below MPDEV intercept.
flag	Placeholder for future development; must be zero.

Return Value

SS$NORMAL	DDT intercept added successfully. This routine may also return various other error status values, including any status returned on a failure to allocate nonpaged pool.

Synchronization Environment

Caller must be in kernel mode, IPL at or below UCB fork IPL.

Almost all use of the DDT within OpenVMS requires holding the UCB fork lock. This is why this routine acquires and conditionally releases the UCB fork lock to change the DDT.

IOC_STD$ESTABLISH_DDT_CANCEL

Establishes DDT$PS_CANCEL_2 intercept.

Functional Description

This routine establishes the intercept of the DDT$PS_CANCEL_2 routine.

Calling Convention

```
int  ioc_std$establish_ddt_cancel (UCB *ucb, int (*cancel_itc_routine)(int chan,
     IRP *irp,PCB *pcb,UCB *ucb, int reason), int level,int flag)
```

Input

ucb	Pointer to a UCB whose DDT$PS_CANCEL_2 is to be intercepted.
cancel_itc_routine	The intercepting `cancel` routine. This routine is called before the driver's `cancel` routine. The calling convention of the `cancel` routine is the same as the standard DDT$PS_CANCEL_2 routine, except that this routine must return one of the following status values: SS$_CHAINW - The next `cancel` routine should be called. SS$_SUPERSEDE - No more `cancel` routines should be called. Any other return value results in a bugcheck.
level	Level of DDT to be intercepted. Currently, multipath does not support an intercept level below MPDEV intercept.
flag	Placeholder for future development; must be zero.

Return Value

SS$NORMAL	DDT intercept added successfully. This routine may also return various other error status values, including any status returned on a failure to allocate nonpaged pool.

Synchronization Environment

Caller must be in kernel mode, IPL at or below UCB fork IPL.

Almost all use of the DDT within OpenVMS requires holding the UCB fork lock. This is why this routine acquires and conditionally releases the UCB fork lock to change the DDT.

IOC_STD$ESTABLISH_DDT_MNTVER

Establishes DDT$PS_MNTVER_2 intercept.

Functional Description

This routine establishes the intercept of the DDT$PS_MNTVER_2 routine.

Calling Convention

```
int  ioc_std$establish_ddt_mntver (UCB *ucb, int(*mntver_itc_routine)(IRP *irp,
     UCB *ucb), int level,int flag)
```

Input

ucb	Pointer to a UCB whose DDT$PS_MNTVER_2 is to be intercepted.
mntver_itc_routine	The intercepting `mntver` routine. This routine is called before the driver's `mntver` routine. The calling convention of the `mntver` routine is the same as the standard DDT$PS_MNTVER_2 routine, except that this routine must return one of the following status values: SS$_CHAINW - The next `mntver` routine should be called. SS$_SUPERSEDE - No more `mntver` routines should be called. Any other return value results in a bugcheck.
level	Level of DDT to be intercepted. Currently, multipath does not support an intercept level below MPDEV intercept.
flag	Placeholder for future development; must be zero.

Return Value

SS$NORMAL	DDT intercept added successfully. This routine may also return various other error status values, including any status returned on a failure to allocate pool.

Synchronization Environment

Caller must be in kernel mode, IPL at or below UCB fork IPL.

Almost all use of the DDT within OpenVMS requires holding the UCB fork lock. This is why this routine acquires and conditionally releases the UCB fork lock to change the DDT.

Device Registration Callback Routines

OpenVMS Alpha Version 7.3-2 also introduces a kernel-mode API for notification of device configuration by way of a callback. This API was designed to enhance the functionality provided by the new IOC_STD$ESTABLISH_DDT_*xxx* routines, which provide a mechanism to intercept calls through a driver dispatch table (DDT).

This API provides a kernel mode "registration" routine, IOC_STD$DEVCONFIG_REGISTER, that privileged code can call, and a complementary routine, IOC_STD$DEVCONFIG_DEREGISTER, to revoke the registration. The registration routine specifies a device class and a callback routine address. Subsequently, when any new device of that class is configured, the specified callback routine is called before the device becomes visible to other threads of execution.

The callback routine can call any of the IOC_STD$ESTABLISH_DDT_*xxx* routines for that device and thus guarantees that the driver intercept is in place before any I/O could possibly be issued to the driver.

IOC_STD$DEVCONFIG_REGISTER

Delivers a notification via a callback when a new device of a specified device class is configured on this system.

Functional Description

IOC_STD$DEVCONFIG_REGISTER is the registration routine that delivers a notification via a callback when a new device of a specified device class is configured on this system. The callback notification occurs when a device is first configured on a system. Notification is not provided when an additional path or a new MSCP server is added for an existing device.

The notification mechanism remains in effect until it is revoked by a call to the IOC_STD$DEVCONFIG_DEREGISTER routine.

Calling Convention

```
int ioc_std$devconfig_register( int flags, int devclass, void (*devconfigured)(UCB
*ucb, int64 user_param), int64 user_param, int64 *ret_handle );
```

Input

flags	Reserved for future enhancements. Must be zero. All other values result in a SS$BADPARAM error.
devclass	The device class value, DC$_*xxx* from devdef.h in STARLET, for which notification is desired. Any value greater than 0 and less than 256 is supported. All other values result in a SS$BADPARAM error.
devconfigured	Address of the caller's desired callback routine, which must be in S0/S1 space. When a new device is configured, this routine is called after the device UCB has been linked into the I/O database and sufficiently initialized so that the I/O database mutex is about to be released. This is after the appropriate driver's structure initialization routine has been called but before the driver's unit initialization is called. The IPL is at the UCB fork IPL, and the UCB fork lock is held.
user_param	Arbitrary 64-bit integer parameter that is passed to the callback routine. Can be used by the callback routine as a context parameter. The same combination of *devclass* value, *devconfigured* value, and *user_param* value cannot be registered twice.

Output

ret_handle	64-bit "handle" that can be used with the `ioc_std$devconfig_deregister` routine to revoke this notification request. The caller should treat the *ret_handle* value as an "opaque" quantity. A r*et_handle* value of zero is returned if the routine fails.

Return Values

SS$_NORMAL	Notification was successfully delivered.
SS$_BADPARAM	The *flags* or *devclass* parameter values are invalid.
SS$_IVADDR	The callback routine address is not in S0/S1 space.

SS$_CBKEXISTS	Callback already exists for this combination of *devclass*, *devconfigured*, and *user_param* values. Multiple registration request for exactly the same notification routine, device class, and parameter are not allowed.
Other return values:	Other error return values are possible, including any error return from an attempt to allocate nonpaged pool.

Synchronization Environment

This routine must be called from kernel mode, process context, IPL 2 or lower. It returns at the entry IPL. This routine declares an SPLIPLHIGH fatal bugcheck if the entry IPL is greater than 2.

Access to the list of registered device configuration callbacks is protected by the I/O database mutex. Therefore, this routine acquires the I/O database mutex for write access and may put the calling process into a resource wait state. This routine releases the I/O database mutex and restore the entry IPL before returning to the caller.

IOC_STD$DEVCONFIG_DEREGISTER

Functional Description

IOC_STD$DEVCONFIG_DEREGISTER revokes a device configuration notification callback that was previously enabled by a call to IOC_STD$DEVCONFIG_REGISTER.

Calling Convention

```
int ioc_std$devconfig_deregister( int64 ret_handle );
```

Input

ret_handle	64-bit "handle" that was returned by a prior call to `ioc_std$devconfig_deregister`.

Return Values

SS$_NORMAL	Successfully revoked notification.
SS$NOSUCHCBK	Did not find a registered device configuration callback with the specified handle, or the handle value is invalid.

Synchronization Environment

This routine must be called from kernel mode, process context, IPL 2 or lower. It returns at the entry IPL. This routine will declare a SPLIPLHIGH fatal bugcheck if the entry IPL is greater than 2.

Access to the list of registered device configuration callbacks is protected by the I/O database mutex. Therefore, this routine acquires the I/O database mutex for write access and may put the calling process into a resource wait state. This routine releases the I/O database mutex before returning to the caller.

Device Configuration Callback Routine

Functional Description

The device configuration callback routine is a caller-specified routine. It is established as a device configuration callback routine by a call to the IOC_STD_$DEVCONFIG_REGISTER register routine.

The device configuration callback routine is called after a new device UCB has been linked into the I/O database and sufficiently initialized such that the I/O database mutex is about to be released. This is after the appropriate driver's structure initialization routine has been called but before the driver's unit init routine is called.

The device configuration routine must be accessible in system context. Therefore, the address of the device configuration routine must be in S0/S1 space. This is enforced by the IOC_STD$DEVCONFIG_REGISTER routine.

The callback is not invoked when an additional path or a new MSCP server is added for an existing device, even though an additional UCB could be created for the new path.

Calling Convention

```
void (*devconfigured)(UCB *ucb, int64 user_param );
```

Input

ucb	Address of the UCB that was just linked into the I/O database.
user_param	64-bit value that was specified on the call to `ioc_std$devconfig_register` that established this callback routine.

Return Values

None.

Synchronization Environment

The device configuration callback routine is called in kernel mode at UCB fork IPL, with the UCB fork lock held. The I/O database mutex is held for write access.

Note that the environment of the device configuration routine is not appropriate for calls to IOC_STD$DEVCONFIG_REGISTER and IOC_STD$DECVCONFIG_DEREGISTER

HP DECdtm Version 2.1

HP DECdtm/XA Version 2.1 Gateway now has clusterwide transaction recovery support. Transactions from applications that use a clusterwide DECdtm Gateway Domain Log can now be recovered from any single-node failure. Gateway servers running on the remaining cluster nodes can initiate the transaction recovery process on behalf of the failed node.

$GETDVI Item Code DVI$SHDW_SITE

The DVI$SHDW_SITE item code returns as a longword the site value for a specified device. The value of DVI$SHDW_SITE is set by the DCL commands SET DEVICE or SET SHADOW.

For more detailed information, refer to the *HP Volume Shadowing for OpenVMS*.

$GETRMI System Service Calls

The new $GETRMI calls provide more robust performance metrics in a number of areas, including overall CPU, memory, MSCP, and distributed lock management.

For more information, refer to the *HP OpenVMS System Services Reference Manual*.

Large-Pool Lookaside Lists

Enhancements to large-pool lookaside lists are being made to increase the lookaside list size. For Gigabit Ethernet devices, this enhancement increases the jumbo receive buffer size and LAN driver buffer size. The enhancement makes overall LAN communication performance more efficient.

For more information, refer to the *HP OpenVMS I/O User's Reference Manual* and the *HP TCP/IP Services for OpenVMS Tuning and Troubleshooting* manual.

POSIX Threads NUMA Support

Starting in OpenVMS Version 7.3-1, the HP POSIX Threads Library can utilize the CPUs in all resource affinity domains (RADs) to execute the threads within a single process. Previously, thread execution was limited essentially to the CPUs within the process's home RAD. Now, when the application workload justifies it, the threads library can create and use kernel threads running on CPUs within additional RADs to execute the application's POSIX threads. One multithreaded process can now utilize all CPUs within a NUMA system.

New System Dump Analyzer (SDA) Commands

The OpenVMS Alpha Version 7.3-2 release includes the new Alpha EV7 On Chip Logic Analyzer (OCLA) utility. With the implementation of this utility, it is now possible to tell which instructions each Alpha EV7 CPU on the system have executed. This is accomplished by setting aside one seventh of the Alpha EV7 cache as acquisition memory, which stores the virtual addresses of instructions executed by the Alpha EV7 CPU. The acquisition memory in the cache can later be analyzed with an SDA extension.

The acquisition of instructions can be enabled or disabled while the system is running, thereby allowing the acquisition of instruction streams for a given period of time without the need to restart the system.

If the OCLA is enabled and started, and your system subsequently fails due to a crash, the current acquisition memory is automatically saved to the system dump file. The instructions executed by each CPU prior to the system failure can then be analyzed with SDA. Upon restart of the system, the acquisition memory in the EV7 is still there and can be copied into system memory using the OCLA ENABLE/OCLA DUMP command.

If the STOP/CPU command is issued on a CPU for which OCLA has been enabled, OCLA automatically is disabled if the CPU is allowed to leave the active set. When a CPU is started with the START/CPU command, OCLA is not automatically enabled; rather, it must be enabled using SDA.

To use the OCLA utility, several new SDA commands and qualifiers are available. The following table describes these SDA commands and qualifiers.

Table 4-1 SDA Commands for the OCLA Utility

Commands	Description
OCLA ENABLE	Enables the OCLA. The command reserves one seventh of the EV7 cache as acquisition memory for instructions. The following qualifiers are available: /CPU=*n* — Specifies the CPU on which to enable OCLA. If this qualifier is omitted, OCLA is enabled on every CPU in the system. /RESET — Resets all OCL registers to known values.
OCLA DISABLE	Disables the OCLA and returns the cache set to the Alpha EV7 CPU. The following qualifiers are available: /CPU=*n* — Specifies the CPU on which to stop acquisition.
OCLA DUMP	Copies the acquisition memory in the Alpha EV7 cache to a region in system space for later analysis by SDA. The following qualifiers are available: /CPU=*n* — Specifies the CPU for which to dump the acquisition memory.

Table 4-1 SDA Commands for the OCLA Utility (Continued)

Commands	Description
OCLA HELP	Provides online help about the OCLA commands.
OCLA LOAD	Loads the OCLA$PCTRACE execlet. This must be done prior to enabling any OCLA. It has no qualifiers.
OCLA SHOW REGISTER	Displays detailed information about the OCLA registers. The following qualifiers are available: /CPU=*n* — Specifies the CPU for which to display registers.
OCLA SHOW STATUS	Displays the status of an OCLA. The following qualifiers are available: /CPU=*n* — Specifies the CPU for which to show OCLA status.
OCLA SHOW TRACE	Decodes the acquired compressed instruction stream and displays it. The following qualifiers are available: /CPU=*n* — Specifies the CPU for which to show data. /LAST*n* — Displays the last *n* instructions. /NOPAL — Do not include PALcode when displaying instructions. /REVERSE — Displays the instructions in reverse order. /SUMMARY — Displays the first 42 instructions. /SYMBOLIZE — Attempts to symbolize each instruction.
OCLA START	Starts the acquisition of instructions into the acquisition memory. The following qualifiers are available: /CPU=*n* — Specifies the CPU on which to start OCLA.
OCLA STOP	Stops the acquisition of instructions. The following qualifiers are available: / CPU=*n* — Specifies the CPU on which to stop acquisition.
OCLA UNLOAD	Unloads the OCLA$PCTRACE execlet and returns the acquisition buffers to the system.

New and Revised System Services

This section contains information about new and changed system services for OpenVMS Alpha Version 7.3-2. The following table summarizes these changes. For more information about these changes, refer to the *HP OpenVMS System Services Reference Manual*.

Table 4-2 New and Revised System Services

System Service	Updated Information
SYS$GOTO_UNWIND_64	New service
SYS$SET_RETURN_VALUE	New service
SYS$GETDVI	The following new item codes have been added:
	DVI$_SHDW_MBR_TWO
	DVI$_SHDW_COPIER_NODE
	DVI$_SHDW_DEVICE_COUNT
	DVI$_SHDW_GENERATION
	DVI$_SHDW_MASTER_MBR
	DVI$_SHDW_MBR_COPY_DONE
	DVI$_SHDW_MBR_COUNT
	DVI$_SHDW_MBR_MERGE_DONE
	DVI$_SHDW_MINIMERGE_ENABLED
	DVI$_SHDW_MBR_READ_COST
	DVI$_SHDW_READ_SOURCE
	DVI$_SHDW_TIMOUT
	DVI$_QLEN
$MOUNT	The MT3 definition MNT$_DENSITY has been added.
$INIT_VOL	The MT3 definition INIT$DENSITY has been added.
$GETRMI	The following new item codes have been added:
	RMI$_CPUEXEC
	RMI$_CPUIDLE
	RMI$_CPUINTSTK
$GETRMI	The following new item codes have been added:
	RMI$_CPUEXEC
	RMI$_CPUIDLE

Table 4-2 New and Revised System Services (Continued)

System Service	Updated Information
	RMI$_CPUINTSTK

New System Services

This section contains information about new system services for OpenVMS Version 7.3-2. For more detailed information, refer to the *HP OpenVMS System Services Reference Manual*.

SYS$GOTO_UNWIND_64 (Alpha and I64 Only)

Description

Unwinds the call stack.

Format

```
SYS$GOTO_UNWIND_64 target_invo ,target_pc ,NewRetVal ,NewRetVal2
```

C prototype

```
int sys$goto_unwind_64(void *target_invo_64, void *(*(target_pc_64), unsigned_int64 *newretval,
unsigned_int64 *newretval2);
```

SYS$SET_RETURN_VALUE (Alpha and I64 Only)

Description

Allows the caller to specify the values in the Mechanism Array and can also be used to set a code condition..

Format

```
SYS$SET_RETURN_VALUE mechanism_arg ,return_type ,return_value
```

C prototype

```
int sys$set_return_value(void *mechanism_arg, unsigned int *return_type, void *return_value);
```

5 Associated Products Features

This chapter describes significant new features of OpenVMS operating system associated products. For a listing and directory information about the OpenVMS associated products, refer to the *Guide to OpenVMS Alpha Version 7.3-2 CD-ROMs*.

The following new features are described:

- ATI RADEON 7500 graphics
- Common Data Security Architecture (CDSA) Version 2.0 enhancements
- OpenVMS e-Business Infrastructure Package
- Kerberos Version 2.0 for OpenVMS
- Secure Sockets Layer (SSL) Version 1.1-A for OpenVMS
- HP TCP/IP Services for OpenVMS

ATI RADEON 7500 Graphics

ATI RADEON 7500 Graphics Support for OpenVMS comprises the 2D and 3D graphics software to support the ATI RADEON 7500 AGP and PCI graphics cards. Numerous features are supported, including the following:

- Use of up to four RADEON cards per system
- Flat panel monitors with a VGA interface
- 3D graphics features
  - — OpenGL Version 1.2 and GLX Version 1.2
  - — Full support of 3D hardware acceleration on all RADEON cards in a system
  - — ATI RADEON 7500 Transformation, Clipping, and Lighting (TCL) hardware
  - — Hardware alpha and stencil planes
  - — 3D multitexturing
  - — 3D cube maps

NOTE You must have an Open3D license to use the 3D hardware-acceleration (direct rendering) features supported by this product.

Multiple head configuration is supported on most Alpha platforms. Refer to the Alpha Systems QuickSpecs web page for the number of cards supported and configuration information.

`http://h18000.www1.hp.com/products/quickspecs/Division/10410.html`

Support for new platforms will be announced on the "What's New" web site when qualification is completed. You can reach this information by clicking on What's new on our site on the following web page:

`http://www.hp.com/go/openvms`

Details about new features are described in the following sections. Also read the ATI RADEON 7500 Graphics releases notes in the *HP OpenVMS Alpha Version 7.3-2 Release Notes* manual.

Multiple RADEON Card Support

Up to four RADEON cards are supported per system. For example, you can have one AGP card with zero to three PCI cards, or you can have one to four PCI cards. All operations, including hardware-accelerated 3D graphics, are supported on every card in the system.

Supported Video Modes

The RADEON 7500 graphics card supports a variety of video modes. It is recommended that you set your refresh rate to the highest value supported by both the RADEON 7500 card and your monitor. Table 5-1 on page 72 lists which video modes are supported by the RADEON 7500 card. All video modes are supported at 8, 16, and 24 bits per pixel. Note that OpenGL rendering is supported only at 24 bits per pixel.

Table 5-1 Supported Video Modes

Monitor Resolution	RADEON 7500 Cards
640x480	60, 72, 75, 85
800x600	60, 72, 75, 85
1024x768	60, 70, 75, 85
1152x864	60
1280x1024	60, 75, 85
1600x1200	60, 65, 75, 85
1920x1440	60, 75
2048x1536	60, 65, 70, 75

The default video mode is 1024x768 at 70 Hz.

To change the default video mode, edit the file SYS$MANAGER:DECW$PRIVATE_SERVER_SETUP.COM. If this file does not exist on your system, you can create it using SYS$MANAGER:DECW$PRIVATE_SERVER_SETUP.TEMPLATE as a model.

Before attempting to change your resolution, make sure that the attached monitor will support the new video mode. For example, to switch to 1280x1024 at 85 Hz, add the following commands to your DECW$PRIVATE_SERVER_SETUP.COM file:

```
$ DECW$XSIZE_IN_PIXELS == "1280"
$ DECW$YSIZE_IN_PIXELS == "1024"
$ DEFINE/SYSTEM/EXECUTIVE/NOLOG DECW$SERVER_REFRESH_RATE "85"
```

For more information about the DECW$PRIVATE_SERVER_SETUP.COM file, refer to *Managing DECwindows Motif for OpenVMS Systems.*

After updating the file, you must either reboot the system or restart the DECwindows server by executing the following command:

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

Monitor Support

The RADEON 7500 card has three built-in connectors:

- An analog VGA connector {labeled CRT on the card)

  Accepts CRT monitors and flat-panel monitors with an analog VGA connector.

- A DVI (digital or analog) video connector

  The DVI-I-to-VGA adapter supplied with the card must always be installed if you are connecting a monitor to the DVI port. You can connect the DVI port to any CRT monitor or flat-panel monitor with an analog VGA connector. Digital output is not supported.

- An S-Video connector

  This is not supported.

Your monitors must already be plugged into the RADEON 7500 card when you power cycle your machine or when you initialize from the system console. Regardless of whether you connect a monitor to the DVI port or the CRT port, you will receive analog video output from each port.

For more information about restrictions, refer to the *HP OpenVMS Alpha Version 7.3-2 Release Notes*.

3D Direct Rendering

If you have an Open3D license, the OpenGL software for the RADEON 7500 graphics card will, by default, force a direct-rendered OpenGL rendering context whenever the OpenGL client requests connection to the DECwindows server using LOCAL transport.

You can override this default behavior and force the default to be indirect rendering by defining the following logical name before running the 3D application:

```
$ DEFINE LIBGL_ALWAYS_INDIRECT 1
```

RADEON 7500 TCL Support

The OpenGL driver for the RADEON 7500 card contains support for the on-board transformation, cli pping, and lighting (TCL) hardware of the RADEON 7500 card for increased primitive rendering performance. You can take advantage of these capabilities if you have an Open3D license. The license is separately orderable (part number QL-0ADA9-AA).

Common Data Security Architecture Version 2.0 Enhancements

Common Data Security Architecture (CDSA) is a multiplatform, industry-standard security infrastructure. It provides a standards-based, stable programming interface that applications can use to access operating system security services, allowing developers to create cross-platform, security-enabled applications. Applications request security services, such as cryptography and other public-key operations, through a

dynamically extensible application programming interface (API). These requests are serviced by a set of plug-in security service provider interfaces (SPIs), which can be supplemented or changed as business needs and technologies evolve.

CDSA V2.0 ships with OpenVMS Version 7.3-2 and is automatically installed with the operating system. CDSA V2.0 Release 3.14 to OpenVMS provides support for all core CDSA functions and includes the following new features:

- Update to OpenSSL V0.9.6G
- Support for signing CDSA applications
- Support for externally developed CDSA plug-ins
- New CDSA example programs for new functionality
- Performance enhancements

To initialize CDSA, log in to the SYSTEM account and enter the following command:

```
$ @SYS$STARTUP:CDSA$UPGRADE.COM
```

You must have SYSPRV to run this procedure. You should also update SYS$MANAGER:SYLOGIN.COM to include the following line:

```
$ @SYS$MANAGER:CDSA$SYMBOLS.COM
```

For complete details about installing and using CDSA, refer to *HP Open Source Security for OpenVMS, Volume 1: Common Data Security Architecture*. The introductory material in this manual has been expanded to include more details about the CDSA architecture, descriptions of the CDSA utility programs, and many more examples. The manual provides the information you need to write your own plug-in for a service provider or a module manager.

OpenVMS e-Business Infrastructure Package

Version 1.5 of the OpenVMS e-Business Infrastructure Package contains the following software components (on two CD-ROMs) to enhance the base OpenVMS Alpha operating system:

- HP BridgeWorks Version 2.1A
- HP OpenVMS Enterprise Directory for e-Business Version 5.2
- HP NetBeans Version 3.5.1
- HP Reliable Transaction Router (RTR) Version 4.2 ECO1
- HP Secure Web Browser Version 1.4 for OpenVMS (based on Mozilla)
- HP Secure Web Server Version 1.3 for OpenVMS (based on Apache)
  - CSWS_Java 2.1 (JServ and Tomcat 4.1.24)
  - CSWS_Perl/Perl 1.1/5.6.1
  - CSWS_PHP 1.1
- Simple Object Access Protocol (SOAP) Version 1.1

- HP Software Development Kit (SDK) Version 1.4.1-2 for the OpenVMS Operating System, *for the Java™ Platform*
- HP XML Technology Version 2.0 for OpenVMS

HP Kerberos Version 2.0 for OpenVMS

The HP Kerberos Version 2.0 for OpenVMS, based on MIT Kerberos V5 Release 1.2.6, is included in the OpenVMS Alpha Version 7.3-2 operating system. Previous versions of OpenVMS included Kerberos V1.0, which was based on MIT Kerberos V5 Release 1.0.5.

Beginning with OpenVMS Alpha Version 7.3-2, Kerberos is no longer an integrated component in the operating system. Kerberos is a required component of OpenVMS, but it is now automatically installed as a layered product.

Kerberos V2.0 includes the following features:

- A fix for the security vulnerability found in earlier versions of Kerberos. For details, see the following web site:

  `http://web.mit.edu/kerberos/advisories/`
- Support for triple DES encryption
- Database enhancements
- DNS support for locating Key Distribution Centers (KDCs)
- Support for new Key Version Number utility (kvno)
- Support for building 32-bit and 64-bit applications (V1.0 supported 64-bit applications only)
- Kerberos V4.0 interoperability
- A new manual, *HP Open Source Security for OpenVMS, Volume 3: Kerberos*, which contains information about installation and configuration, client progrsms, a programming tutorial, and reference information about the GSSAPI and KRB5 application programming interface

Kerberos is a network authentication protocol designed to provide strong authentication for client/server applications by using secret-key cryptography.

Kerberos was created by the Massachusetts Institute of Technology as a solution for network security. The Kerberos protocol uses strong cryptography so that a client can prove its identity to a server, and a server can prove its identity to a client, across an insecure network connection. After a client and server have used Kerberos to prove their identity, they can also encrypt all of their communications to ensure privacy and data integrity.

For more information, refer to *HP Open Source Security for OpenVMS, Volume 3: Kerberos* and the Kerberos for OpenVMS web site at:

`http://h71000.www7.hp.com/openvms/products/kerberos/`

HP Secure Sockets Layer (SSL) for OpenVMS

HP SSL Version 1.1-A for OpenVMS, based on OpenSSL 0.9.6G, is included with the OpenVMS Alpha Version 7.3-2 operating system. The previous version of OpenVMS included SSL V1.0, which was based on OpenSSL 0.9.6B.

New features in SSL Version 1.1-A include:

- A port of the Open SSL 0.9.6G baselevel, which includes fixes to reported security vulnerabilities:
  - `http://www.openssl.org/news/secadv_20030930.txt`
  - `http://www.openssl.org/news/secadv_20020730.txt`
  - `http://www.openssl.org/news/secadv_20030219.txt`
  - `http://www.openssl.org/news/secadv_20030317.txt`
  - `http://www.openssl.org/news/secadv_20030319.txt`
- Certificate Revocation List (CRL) support in the Certificate Tool
- A DES encryption image that allows you to enable uuencoding and uudecoding

Secure Sockets Layer (SSL) was first developed by Netscape to secure browser to web server communication channels. OpenSSL is an industry standard package of cryptographic services that provide SSL and Transaction Layer Security (TLS) protocols. Additionally, Open SSL provides client/server authentication, encryption, and message integrity services for secure communications on the TCP/IP protocol.

For more information, refer to *HP Open Source Security for OpenVMS, Volume 2: HP SSL for OpenVMS* and the SSL for OpenVMS web site at:

`http://h71000.www7.hp.com/openvms/products/ssl/`

HP TCP/IP Services for OpenVMS

TCP/IP Services for OpenVMS Version 5.4 provides performance enhancements as well as new features. TCP/IP Services Version 5.4 is supported on OpenVMS Alpha Versions 7.3-1 and 7-3.2 systems only.

The new features and enhancements to TCP/IP Services Version 5.4 include:

- A new kernel that provides performance scalability for SMP systems (requires OpenVMS Alpha Version 7.3-2)
- Secure shell (SSH) client and server
- Secure Sockets Layer (SSL) for POP
- failSAFE IP (IP address failover)
- Software Update and new programming examples using IPv6 APIs
- BIND server version 9.2.1
- INET driver performance enhancements
- NFS server performance enhancements

- TELNET server performance enhancements
- BG device creation enhancement
- Fast BG device creation and deletion
- Updated standard kernel
- Support for the `tcpdump` utility

For more information about these changes to TCP/IP Services, refer to the *HP TCP/IP Services for OpenVMS Version 5.4 Release Notes*.

II OpenVMS Documentation

This part contains information about documentation changes and documentation sets for OpenVMS Alpha Version 7.3-2.

6 OpenVMS Documentation Overview

Table 6-1 describes any reorganization of the OpenVMS documentation set for OpenVMS Alpha Version 7.3-2. For this release, there is one new manual and one archived manual.

Table 6-1 Documentation Set Changes for OpenVMS Alpha Version 7.3-2

Manual	Change	Description
HP Open Source Security for OpenVMS, Volume 3: Kerberos	New	This new manual for Version 7.3-2 is for application developers who want to implement the Kerberos protocol that uses strong cryptography, so that a client can prove its identity to a server across a nonsecure network connection.
COM, Registry, and Events for OpenVMS Developer's Manual	Archived	Beginning with Version 7.3-2, this manual is archived. The online version is included with other archived manuals in a separate directory on the OpenVMS documentation CD-ROMs. The online version and information about COM for OpenVMS can be found at the following web site: `http://www.hp.com/products/openvms/com/`
OpenVMS Management Station Overview and Release Notes	Archived	Beginning with Version 7.3-2, this manual is archived. The online version has been updated to include release notes that describe information regarding OpenVMS Management Station Version 3.2-B. You will find the online version on the OpenVMS documentation CD-ROMS and also on the OpenVMS documentation web site: `http://www.hp.com/go/openvms/doc/`

7 OpenVMS Printed and Online Documentation

OpenVMS documentation is provided in the following formats:

- Printed documentation

  If you need hardcopy documents, you can purchase most OpenVMS manuals as printed books.

- Online documentation on CD-ROM

  All OpenVMS manuals are available in online formats on CD-ROMs that also include the documentation for many associated products. You automatically receive the documentation CD-ROMs in your OpenVMS media kit.

- Online documentation on the OpenVMS web site

  You can view any OpenVMS document, including archived manuals, on the OpenVMS web site.

  `http://www.hp.com/go/openvms/doc`

- Online help

  You can quickly display online help for OpenVMS commands, utilities, and system routines when you need task-related information.

The following sections describe each format in which OpenVMS documentation is provided and specifies the titles that are available in that format.

Printed Documentation

Some printed documentation comes with your OpenVMS media kit. All other printed manuals are separately orderable, either in kits or individually. This section describes the OpenVMS printed documentation offerings, which are broken down into the following categories:

- Media kit
- Documentation sets: Base Set and Full Set
- System Integrated Products (SIPs)
- Archived manuals

OpenVMS Media Kit Documentation

The OpenVMS media kit contains the documents you need to get started with the latest version of the OpenVMS operating system. Table 7-1 lists the books included in the OpenVMS media kit. The books you receive are determined by whether you are a new or a service customer. New customers receive all books: service customers receive only new books and books that have been updated since the last release. You can also order the manuals listed in Table 7-1 individually.

Table 7-1 OpenVMS Media Kit Documentation

Manual	Order Number
OpenVMS License Management Utility Manual[1]	AA-PVXUF-TK
Guide to OpenVMS Alpha Version 7.3-2 CD-ROMs	AA-RV90A-TE
HP OpenVMS Alpha Version 7.3-2 New Features and Documentation Overview	AA-RV8XA-TE
HP OpenVMS Alpha Version 7.3-2 Upgrade and Installation Manual	AA-RV8ZA-TE
HP OpenVMS Alpha Version 7.3-2 Release Notes	AA-RV8YA-TE

[1] Not revised for Version 7.3-2.

OpenVMS Documentation Sets

Table 7-2 describes the OpenVMS documentation that is available in the following documentation sets.

Table 7-2 OpenVMS Documentation Sets

Documentation Set	Description	Order Number
Full set	Intended for users who need extensive explanatory information for all major OpenVMS resources. Contains all the OpenVMS documentation in one offering. Includes the Base Documentation set.	QA-001AA-GZ.7.3-2
Base set	Subset of the Full Documentation set. Intended for general users and system managers of small standalone systems. Includes the most commonly used OpenVMS manuals.	QA-09SAA-GZ-7.3-2

In addition to ordering the Full or Base Documentation set, you can order any OpenVMS manual individually.

Table 7-3 lists the manuals in the OpenVMS Base and Full Documentation sets. For a description of each manual, see Chapter 8.

Table 7-3 OpenVMS Full Documentation Set (QA-001AA-GZ.7.3-2)

Manual	Order Number
OpenVMS Base Documentation Set	**QA-09SAA-GZ.7.3-2**
HP OpenVMS DCL Dictionary: A-M[1]	AA-PV5KJ-TK
HP OpenVMS DCL Dictionary: N-Z[1]	AA-PV5LJ-TK
HP OpenVMS Guide to System Security[1]	AA-Q2HLG-TE
HP OpenVMS System Management Utilities Reference Manual: A-L[1]	AA-PV5PH-TK
HP OpenVMS System Management Utilities Reference Manual: M-Z[1]	AA-PV5QH-TK
HP OpenVMS System Manager's Manual, Volume 1: Essentials[1]	AA-PV5MH-TK
HP OpenVMS System Manager's Manual, Volume 2: Tuning, Monitoring, and Complex Systems[1]	AA-PV5NH-TK
OpenVMS User's Manual	AA-PV5JG-TK
HP OpenVMS Alpha Version 7.3-2 New Features and Documentation Overview[2]	AA-RV8XA-TE
HP OpenVMS Alpha Version 7.3-2 Release Notes[2]	AA-RV8YA-TE
Additional Books Included in the Full Documentation Set	**QA-001AA-GZ.7.3-2**
Availability Manager User's Manual	AA-RNSJB-TE
HP C Run-Time Library Reference Manual for OpenVMS Systems[1]	AA-RSMUB-TE
C Run-Time Library Utilities Reference Manual	AA-R238C-TE
Compaq Portable Mathematics Library	AA-PV6VE-TE
DECamds User's Guide	AA-Q3JSE-TE
DEC Text Processing Utility Reference Manual	AA-PWCCD-TE
Extensible Versatile Editor Reference Manual	AA-PWCDD-TE
Guidelines for OpenVMS Cluster Configurations[1]	AA-Q28LG-TK
Guide to Creating OpenVMS Modular Procedures	AA-PV6AD-TK
Guide to OpenVMS File Applications	AA-PV6PE-TK
Guide to the POSIX Threads Library	AA-QSBPD-TE
Guide to DEC Text Processing Utility	AA-PWCBD-TE

Table 7-3 OpenVMS Full Documentation Set (QA-001AA-GZ.7.3-2) (Continued)

Manual	Order Number
HP Open Source Security for OpenVMS, Volume 1: Common Data Security Architecture[1]	AA-RSCUB-TE
HP Open Source Security for OpenVMS, Volume 2: HP SSL for OpenVMS[1]	AA-RSCVB-TE
HP Open Source Security for OpenVMS, Volume 3: Kerberos[2]	AA-RUEBA-TE
HP OpenVMS Alpha Partitioning and Galaxy Guide[1]	AA-REZQE-TE
OpenVMS Alpha Guide to Upgrading Privileged-Code Applications	AA-QSBGD-TE
OpenVMS Alpha System Analysis Tools Manual	AA-REZTC-TE
OpenVMS Calling Standard	AA-QSBBD-TE
OpenVMS Cluster Systems	AA-PV5WF-TK
OpenVMS Command Definition, Librarian, and Message Utilities Manual	AA-QSBDD-TE
OpenVMS Debugger Manual	AA-QSBJD-TE
OpenVMS Delta/ XDelta Debugger Manual	AA-PWCAD-TE
HP OpenVMS I/ O User's Reference Manual[1]	AA-PV6SF-TK
OpenVMS Linker Utility Manual	AA-PV6CD-TK
OpenVMS MACRO-32 Porting and User's Guide	AA-PV64D-TE
OpenVMS Performance Management	AA-R237C-TE
HP OpenVMS Programming Concepts Manual, Volume I	AA-RNSHB-TE
HP OpenVMS Programming Concepts Manual, Volume II	AA-PV67F-TK
OpenVMS Record Management Services Reference Manual	AA-PV6RE-TK
OpenVMS Record Management Utilities Reference Manual	AA-PV6QD-TK
OpenVMS RTL General Purpose (OTS$) Manual	AA-PV6HD-TK
OpenVMS RTL Library (LIB$) Manual	AA-QSBHD-TE
OpenVMS RTL Screen Management (SMG$) Manual	AA-PV6LD-TK
OpenVMS RTL String Manipulation (STR$) Manual	AA-PV6MD-TK
OpenVMS System Messages: Companion Guide for Help Message Users	AA-PV5TD-TK
HP OpenVMS System Services Reference Manual: A-GETUAI[1]	AA-QSBMF-TE
HP OpenVMS System Services Reference Manual: GETUTC-Z[1]	AA-QSBNF-TE

Table 7-3 OpenVMS Full Documentation Set (QA-001AA-GZ.7.3-2) (Continued)

Manual	Order Number
OpenVMS Utility Routines Manual	AA-PV6EF-TK
OpenVMS VAX RTL Mathematics (MTH$) Manual	AA-PVXJD-TE
OpenVMS VAX System Dump Analyzer Manual	AA-PV6TD-TE
HP POLYCENTER Software Installation Utility Developer's Guide	AA-Q28ME-TK
VAX MACRO and Instruction Set Reference Manual	AA-PS6GD-TE
HP Volume Shadowing for OpenVMS[1]	AA-PVXMJ-TE

[1] Revised for Version 7.3-2.
[2] New for Version 7.3-2.

Documentation for System Integrated Products

System Integrated Products (SIPs) are included in the OpenVMS software, but you must purchase separate licenses to enable them. Table 7-4 lists the documentation associated with System Integrated Products.

Table 7-4 System Integrated Products Documentation

System Integrated Product	Related Documentation
Galaxy Software Architecture on OpenVMS Alpha	The documentation is included in the OpenVMS Full Documentation set, or you can order it separately.
OpenVMS Clusters	The OpenVMS Cluster documentation is included in the OpenVMS Full Documentation set.
RMS Journaling for OpenVMS	You must purchase the RMS Journaling documentation separately. Use the following part number: *RMS Journaling for OpenVMS Manual* (AA-JG41C-TE).
Volume Shadowing for OpenVMS	The documentation is included in the OpenVMS Full Documentation set, or you can order it separately.

Archived OpenVMS Documentation

OpenVMS continually updates, revises, and enhances the OpenVMS operating system documentation. From time to time, manuals are archived. You can access the archived manuals online from the documentation CD-ROMs or from the following web site:

`http://www.hp.com/go/openvms/doc/`

For a list of the archived OpenVMS manuals, see "Archived Manuals" on page 97.

New Authoring Tool for OpenVMS Documentation

With OpenVMS Alpha Version 7.3-2, the OpenVMS Documentation team is continuing to introduce books that have been authored and published using a tool based on the Standard Generalized Markup Language (SGML). SGML, a standard within the industry, provides a number of benefits to customers and to OpenVMS documentation.

Readers will notice a difference in appearance between books produced from SGML and others in the documentation set. This is true for HTML, PDF, and printed formats and is a natural result of the new authoring environment.

The following Version 7.3-2 manuals were authored using the new tool:

- *HP OpenVMS Alpha Partitioning and Galaxy Guide*
- *HP OpenVMS Alpha Version 7.3-2 New Features and Documentation Overview*
- *HP OpenVMS Alpha Version 7.3-2 Upgrade and Installation Manual*
- *Guide to OpenVMS Alpha Version 7.3-2 CD-ROMs*
- *HP Open Source Security for OpenVMS, Volume 1: Common Data Security Architecture*
- *HP Open Source Security for OpenVMS, Volume 2: HP SSL for OpenVMS*
- *HP Open Source Security for OpenVMS, Volume 3: Kerberos*
- *HP OpenVMS I/O User's Reference Manual*
- *HP OpenVMS System Manager's Manual, Volume 1: Essentials*
- *HP OpenVMS System Manager's Manual, Volume 2: Tuning, Monitoring, and Complex Systems*
- *HP TCP/IP Services for OpenVMS Guide to SSH*
- *HP Volume Shadowing for OpenVMS*

Online Documentation CD-ROM

Online documentation for the OpenVMS operating system and many associated products is provided on two CD-ROMs. One is an ISO 9660 Level 2 CD-ROM that is readable on Windows® and Macintosh® systems. The other is a Files-11 CD-ROM that is readable on Windows and OpenVMS systems. The contents of the CD-ROMs are the same except for the following:

- The ISO 9660 Level 2 CD-ROM contains Adobe® Acrobat® Reader Version 5.0.5.
- The Files-11 CD-ROM contains the HP Secure Web Browser for OpenVMS Alpha (based on Mozilla) and a command procedure to launch the browser.

Online Formats

The documentation CD-ROMs contain documentation in the following formats:

Documentation	Available Formats
Current OpenVMS manuals	HTML, PDF
Archived manuals	PDF
HP OpenVMS Alpha Version 7.3-2 Upgrade and Installation Manual	HTML, PDF, ASCII text
HP OpenVMS Alpha Version 7.3-2 Release Notes	HTML, PDF, ASCII text
HP OpenVMS Alpha Version 7.3-2 New Features and Documentation Overview	HTML, PDF, ASCII text
Layered product documents (release notes, installation guides, Software Product Descriptions, cover letters)	PostScript®, ASCII text

Bookreader files are no longer available on the documentation CD-ROMs.

PDF Reader

The Adobe Acrobat Reader is provided for viewing PDF files. This self-extracting file can be installed on a computer running Windows. It is located on the ISO 9660 Level 2 CD-ROM.

For information about how to access documents on the documentation CD-ROMs and about the PDF reader, refer to the *Guide to OpenVMS Alpha Version 7.3-2 CD-ROMs*.

Online Documentation on the OpenVMS Web Site

You can access OpenVMS manuals in various online formats from the following OpenVMS web site:

`http://www.hp.com/go/openvms/doc/`

This site contains links to current versions of manuals in the OpenVMS Full Documentation set as well as to manuals for selected layered products.

Online Help

The OpenVMS operating system provides online help for the commands, utilities, and system routines documented in the Full Documentation set.

You can use the Help Message facility to quickly access online descriptions of system messages. In addition, you can add your own source files, such as messages documentation that you have written to the Help Message database. The *OpenVMS System Messages: Companion Guide for Help Message Users* manual explains how to use the Help Message facility. You can also access DCL Help for Help Message by entering:

```
$ HELP HELP/MESSAGE
```

Reference information for the following OpenVMS utility routines is now included in online help under RTL_ROUTINES:

Access Control List (ACL) Editor (ACLEDIT$)
Backup (BACKUP) (BACKUP$)
Command Language (CLI) (CLI$)
Command File Qualifier (UTIL$)
Convert (CONVERT) (CONV$)
Data Compression/Expansion (DCX) (DCX$)
DEC Text Processing Utility (DECTPU) (TPU$)
EDT (EDT$EDIT; FILEIO; WORKIO; XLATE)
File Definition Language (FDL) (FDL$)
Librarian (LBR) (LBR$)
LOGINOUT (LGI) (LGI$)
Mail Utility (MAIL) (MAIL$)
National Character Set (NCS) (NCS$)
Print Symbiont Modification (PSM) (PSM$; USER-*x*)
Symbiont/Job Controller Interface (SMB) (SMB$)
Sort/Merge (SOR) (SOR$)

8 Descriptions of OpenVMS Manuals

This chapter provides summary descriptions for the following OpenVMS documentation:

- Manuals in the OpenVMS Media Kit
- Manuals in the OpenVMS Base and Full Documentation sets
- OpenVMS Alpha Device Driver manual
- RMS Journaling manual
- Archived manuals

Manuals in the OpenVMS Media Kit

Guide to OpenVMS Alpha Version 7.3-2 CD-ROMs

Provides information about the OpenVMS Alpha operating system and documentation CD-ROMs. Lists the contents of the OpenVMS Version 7.3-2 CD-ROM package, includes pointers to installation information, and gives instructions about how to access manuals on the documentation CD-ROM.

OpenVMS License Management Utility Manual

Describes the License Management Facility (LMF), the OpenVMS license management tool. LMF includes the License Management Utility (LICENSE) and VMSLICENSE.COM, the command procedure you use to register, manage, and track software licenses.

HP OpenVMS Alpha Version 7.3-2 Upgrade and Installation Manual

Provides step-by-step instructions for installing and upgrading the OpenVMS Alpha operating system on Alpha computers. Includes information about booting, shutdown, backup, and licensing procedures.

HP OpenVMS Alpha Version 7.3-2 New Features and Documentation Overview

Describes new and improved components for the Alpha operating system for the Version 7.3-2 release. Includes information about OpenVMS documentation changes for Version 7.3-2 as well as the printed and online OpenVMS documentation offerings.

HP OpenVMS Alpha Version 7.3-2 Release Notes

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

Manuals in the Base Documentation Set

HP OpenVMS DCL Dictionary

Describes the DIGITAL Command Language (DCL) and provides an alphabetical listing of detailed reference information and examples for all DCL commands and lexical functions. This manual is in two volumes.

HP OpenVMS Guide to System Security

Describes the security features available in the OpenVMS Alpha and VAX operating systems. Explains the purpose and proper application of each feature in the context of specific security needs.

HP OpenVMS System Management Utilities Reference Manual

Presents reference information about the utilities you can use to perform system management tasks on your system as well as the tools to control and monitor system access and resources. Includes a description of the AUTOGEN command procedure. This manual is in two volumes.

HP OpenVMS System Manager's Manual, Volume 1: Essentials

Provides instructions for setting up and maintaining routine operations such as starting up the system, installing software, and setting up print and batch queues. Also, explains routine disk and magnetic tape operations.

HP OpenVMS System Manager's Manual, Volume 2: Tuning, Monitoring, and Complex Systems

Describes how to configure and control the network, how to monitor the system, and how to manage system parameters. Also, includes information about OpenVMS Cluster systems, network environments, and DECdtm functionality.

OpenVMS User's Manual

Provides an overview of the operating system and presents basic concepts, task information, and reference information that allow you to perform daily computing tasks. Describes how to work with files and directories. Also includes these additional topics:

- Sending messages with the Mail utility and the Phone utility
- Using the Sort/Merge utility
- Using logical names and symbols
- Writing command procedures
- Editing files with the EVE and EDT text editors

HP OpenVMS Alpha Version 7.3-2 New Features and Documentation Overview

Describes new and improved components for the Alpha operating system for Version 7.3-2 release. Includes information about OpenVMS documentation changes for Version 7.3-2 as well as the printed and online OpenVMS documentation offerings.

HP OpenVMS Alpha Version 7.3-2 Release Notes

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

Additional Manuals in the OpenVMS Full Documentation Set

Availability Manager User's Manual

Describes how to use the HP Availability Manager system management tool, from either an OpenVMS Alpha or a Windows node, to monitor one or more OpenVMS nodes on an extended local area network (LAN) or to target a specific node or process for detailed analysis.

HP C Run-Time Library Reference Manual for OpenVMS Systems

Provides reference information on the functions and macros found in the HP C RTL that perform I/O operations, character and string manipulation, mathematical operations, error detection, subprocess creation, system access, and screen management. Includes portability concerns between operating systems, and describes the HP C for OpenVMS socket routines used for writing Internet application programs for the TCP/IP protocol.

C Run-Time Library Utilities Reference Manual

Provides detailed usage and reference information about Compaq C Run-Time Library utilities for managing localization and time zone data in international software applications.

Compaq Portable Mathematics Library

Documents the mathematics routines in the Compaq Portable Mathematics Library (DPML), supplied only with the OpenVMS Alpha systems. VAX programmers should refer to the *OpenVMS VAX RTL Mathematics (MTH$) Manual.*

DECamds User's Guide

Provides information for installing and using the DECamds software. DECamds is a system management tool that lets you monitor, diagnose, and track events in OpenVMS system and OpenVMS Cluster environments.

DEC Text Processing Utility Reference Manual

Describes the DEC Text Processing utility (DECTPU) and provides reference information about the EDT Keypad Emulator interfaces to DECTPU.

Extensible Versatile Editor Reference Manual

Contains command reference information about the EVE text editor. Also provides a cross-reference between EDT and EVE commands.

Guidelines for OpenVMS Cluster Configurations

Provides information to help you choose systems, interconnects, storage devices, and software. It can help you configure these components to achieve high availability, scalability, performance, and ease of system management. Detailed directions using SCSI and Fibre Channel in an OpenVMS Cluster system are also included in this manual.

Guide to Creating OpenVMS Modular Procedures

Describes how to perform a complex programming task by dividing it into modules and coding each module as a separate procedure.

Guide to OpenVMS File Applications

Contains guidelines for designing, creating, and maintaining efficient data files by using Record Management Services (RMS). This manual is intended for application programmers and designers responsible for programs that use RMS files, especially if performance is an important consideration.

Guide to the POSIX Threads Library

Describes the POSIX Threads Library (formerly named DECthreads) package, HP's multithreading run-time libraries. Use the routines in this package to create and control multiple threads of execution within the address space provided by a single process. Offering both usage tips and reference synopses, this document describes three interfaces: routines that conform to the IEEE POSIX 1003.1c standard (called *pthread*), routines that provide thread-related services in nonthreaded applications (called thread-independent services or *tis*), and a set of HP proprietary routines (called *cma*) that provide a stable, upward-compatible interface.

Guide to the DEC Text Processing Utility

Provides an introduction to developing DECTPU programs.

HP Open Source Security for OpenVMS, Volume 1: Common Data Security Architecture

For application developers who want to use the Common Data Security Architecture (CDSA) to add security to their programs. Describes CDSA, gives information about installation and initialization, and provides example programs. Contains the CDSA application programming interface modules.

HP Open Source Security for OpenVMS, Volume 2: HP SSL for OpenVMS

For application developers who want to protect communication links to OpenVMS applications with HP Secure Sockets Layer (HP SSL) for OpenVMS Alpha. Contains installation instructions, release notes, and provides example programs. Includes programming information and a reference section for the OpenSSL application programming interface modules.

HP Open Source Security for OpenVMS, Volume 3: Kerberos

For application developers who want to implement the Kerberos protocol that uses strong cryptography, so that a client can prove its identity to a server (and vice versa) across an insecure network connection.

HP OpenVMS Alpha Partitioning and Galaxy Guide

Provides complete details about how to use all of the OpenVMS Galaxy features and capabilities available in OpenVMS Alpha Version 7.3-2. Includes procedures for creating, managing, and using OpenVMS Galaxy computing environments on AlphaServer systems.

OpenVMS Alpha Guide to Upgrading Privileged-Code Applications

Explains the OpenVMS Alpha Version 7.0 changes that might impact Alpha privileged-code applications and device drivers as a result of the OpenVMS Alpha 64-bit virtual addressing and kernel threads support provided in OpenVMS Alpha Version 7.0

Privileged-code applications from versions prior to OpenVMS Alpha Version 7.0 might require the source-code changes described in this guide.

OpenVMS Alpha System Analysis Tools Manual

Describes the following system analysis tools in detail, while also providing a summary of the dump off system disk (DOSD) capability and the DELTA/XDELTA debugger:

- System Dump Analysis (SDA)
- System code debugger (SCD)
- System dump debugger (SDD)
- Watchpoint utility

Intended primarily for the system programmer who must investigate the causes of the system failures and debug kernel mode code, such as a device driver.

OpenVMS Calling Standard

Documents the calling standard for the OpenVMS Alpha and VAX operating systems.

OpenVMS Cluster Systems

Describes procedures and guidelines for configuring and managing OpenVMS Cluster systems. Also describes how to provide high availability, building-block growth, and unified system management across clustered systems.

OpenVMS Command Definition, Librarian, and Message Utilities Manual

Contains descriptive and reference information about the following utilities:

- Command Definition utility
- Librarian utility
- Message utility

OpenVMS Debugger Manual

Explains the features of the OpenVMS Debugger for programmers.

OpenVMS Delta/XDelta Debugger Manual

Describes the Delta/XDelta utility used to debug programs that run in privileged processor mode or at an elevated interrupt priority level.

HP OpenVMS I/O User's Reference Manual

Contains the information that system programmers need to program I/O operations using the device drivers that are supplied with the operating system.

OpenVMS Linker Utility Manual

Describes how to use the Linker utility to create images that run on OpenVMS systems. Also explains how to control a link operation with link qualifiers and link options.

OpenVMS MACRO-32 Porting and User's Guide

Describes how to port existing VAX MACRO assembly language code to an OpenVMS Alpha system by using the features of the MACRO-32 compiler. Also documents how to use the compiler's 64-bit addressing support.

OpenVMS Performance Management

Introduces and explains the techniques used to optimize performance on an OpenVMS system.

OpenVMS Programming Concepts

Describes concepts such as process creation, kernel threads, and the kernel threads process structure, interprocess communications, process control, data sharing, condition handling, and ASTs. This two-volume manual uses system services, utility routines, and run-time library (RTL) routines to illustrate mechanisms for utilizing OpenVMS features.

OpenVMS Record Management Services Reference Manual

Provides reference and usage information for all programmers who use RMS data files.

OpenVMS Record Management Utilities Reference Manual

Contains descriptive and reference information about the following RMS utilities:

- Analyze/RMS_File utility
- Convert and Convert/Reclaim utilities

- File Definition Language facility

OpenVMS RTL General Purpose (OTS$) Manual

Documents the general-purpose routines contained in the OTS$ facility of the OpenVMS Run-Time Library. Indicates which routines are specific to Alpha or VAX, as well as how routines function differently on each system.

OpenVMS RTL Library (LIB$) Manual

Documents the general-purpose routines contained in the LIB$ facility of the OpenVMS Run-Time Library. Indicates which routines are specific to Alpha or VAX, as well as how routines function differently on each system.

OpenVMS RTL Screen Management (SMG$) Manual

Documents the screen management routines contained in the SMG$ facility of the OpenVMS Run-Time Library. Indicates which routines are specific to Alpha or VAX, as well as how routines function differently on each system.

OpenVMS RTL String Manipulation (STR$) Manual

Documents the string manipulation routines contained in the STR$ facility of the OpenVMS Run-Time Library. Indicates which routines are specific to Alpha or VAX, as well as how routines function differently on each system.

OpenVMS System Messages: Companion Guide for Help Message Users

Describes features of the Help Message facility, a tool that you can use to display message descriptions. Describes the HELP/MESSAGE command and qualifiers and also includes detailed information about customizing the Help Message database. Also provides descriptions of message that can occur when the system and Help Message are not fully operable.

HP OpenVMS System Services Reference Manual

Presents the set of routines that the operating system uses to control resources, allow process communications, control I/O, and perform other such operating system functions. This manual is in two volumes.

OpenVMS Utility Routines Manual

Describes the routines that allow a program to use the callable interface of selected OpenVMS utilities.

OpenVMS VAX RTL Mathematics (MTH$) Manual

Documents the mathematics routines contained in the MTH$ facility of the OpenVMS Run-Time Library, which is relevant only to programmers using OpenVMS VAX. (Alpha programmers should refer to *Compaq Portable Mathematics Library*.)

OpenVMS VAX System Dump Analyzer Manual

Explains how to use the System Dump Analyzer utility to investigate system failures and examine a running OpenVMS VAX system. VAX programmers should refer to this manual; Alpha programmers should refer to the *OpenVMS Alpha System Dump Analyzer Utility Manual*.

HP POLYCENTER Software Installation Utility Developer's Guide

Describes the procedure and provides guidelines for developing software products that will be installed using the POLYCENTER Software Installation utility. Intended for developers who are designing installation procedures for software products layered on the OpenVMS operating system.

VAX MACRO and Instruction Set Reference Manual

Documents both the assembler directives of VAX MACRO and the VAX instruction set.

HP Volume Shadowing for OpenVMS

Describes how to provide high data availability with phase II volume shadowing.

RMS Journaling Manual

RMS Journaling for OpenVMS

Describes the three types of RMS Journaling as well as other OpenVMS components that support RMS Journaling. This manual also describes the RMS Recovery utility (which is used to recover data saved using journaling), the transaction processing system services, and system management tasks required when using RMS Journaling.

Archived Manuals

Table 8-1 lists the OpenVMS manuals that have been archived. Please note the following:

- Most archived manuals can be ordered separately in printed format from HP.
- Archived manuals are available in various formats (PostScript or PDF) on the documentation CD-ROMs.
- Most information from the archived manuals has been incorporated in other documents or online help.

Table 8-1 Archived OpenVMS Manuals

Manual	Order Number
A Comparison of System Management on OpenVMS AXP and OpenVMS VAX	AA-PV71B-TE
Building Dependable Systems: The OpenVMS Approach	AA-PV5YB-TE
COM, Registry, and Events for OpenVMS Developer's Manual	AA-RSCWA-TE
Creating an OpenVMS Alpha Device Driver from an OpenVMS VAX Device Driver	AA-R0Y8A-TE
Creating an OpenVMS AXP Step 2 Device Driver from a Step 1 Device Driver	AA-Q28TA-TE
Creating an OpenVMS AXP Step 2 Device Driver from an OpenVMS VAX Device Driver	AA-Q28UA-TE
Guide to OpenVMS AXP Performance Management	AA-Q28WA-TE
Guide to OpenVMS Performance Management	AA-PV5XA-TE

Table 8-1 Archived OpenVMS Manuals (Continued)

Manual	Order Number
Migrating an Application from OpenVMS VAX to OpenVMS Alpha	AA-KSBKB-TE
Migrating an Environment from OpenVMS VAX to OpenVMS Alpha	AA-QSBLA-TE
Migrating to an OpenVMS AXP System: Planning for Migration	AA-PV62A-TE
Migrating to an OpenVMS AXP System: Recompiling and Relinking	AA-PV63A-TE
OpenVMS Alpha Guide to 64-Bit Addressing and VLM Features	AA-QSBCC-TE
OpenVMS Alpha System Dump Analyzer Utility Manual	AA-PV6UC-TE
OpenVMS AXP Device Support: Developer's Guide	AA-Q28SA-TE
OpenVMS AXP Device Support: Reference	AA-Q28PA-TE
OpenVMS Bad Block Locator Utility Manual	AA-PS69A-TE
OpenVMS Compatibility Between VAX and Alpha	AA-PYQ4C-TE
OpenVMS Developer's Guide to VMSINSTAL	AA-PWBXA-TE
OpenVMS DIGITAL Standard Runoff Reference Manual	AA-PS6HA-TE
OpenVMS EDT Reference Manual	AA-PS6KA-TE
OpenVMS Exchange Utility Manual	AA-PS6AA-TE
OpenVMS Glossary	AA-PV5UA-TK
OpenVMS Guide to Extended File Specifications	AA-REZRB-TE
OpenVMS Management Station Overview and Release Notes	AA-QJGCE-TE
OpenVMS Master Index	AA-QSBSD-TE
OpenVMS National Character Set Utility Manual	AA-PS6FA-TE
OpenVMS Obsolete Features Manual	AA-PS6JA-TE
OpenVMS Programming Environment Manual	AA-PV66B-TK
OpenVMS Programming Interfaces: Calling a System Routine	AA-PV68B-TK
OpenVMS RTL DECtalk (DTK$) Manual	AA-PS6CA-TE
OpenVMS RTL Parallel Processing (PPL$) Manual	AA-PV6JA-TK
OpenVMS Software Overview	AA-PVXHB-TE
OpenVMS SUMSLP Utility Manual	AA-PS6EA-TE

Table 8-1 Archived OpenVMS Manuals (Continued)

Manual	Order Number
OpenVMS System Messages and Recovery Procedures Reference Manual: A–L	AA-PVXKA-TE
OpenVMS System Messages and Recovery Procedures Reference Manual: M–Z	AA-PVXLA-TE
OpenVMS Terminal Fallback Utility Manual	AA-PS6BA-TE
OpenVMS VAX Card Reader, Line Printer, and LPA11–K I/O User's Manual	AA-PVXGA-TE
OpenVMS VAX Device Support Manual	AA-PWC8A-TE
OpenVMS VAX Device Support Reference Manual	AA-PWC9A-TE
OpenVMS VAX Patch Utility Manual	AA-PS6DA-TE
OpenVMS Wide Area Network I/O User's Reference Manual	AA-PWC7A-TE
PDP-11 TECO User's Guide	AA-K420B-TC
POLYCENTER Software Installation Utility User's Guide	AA-Q28NA-TK
Standard TECO Text Editor and Corrector for the VAX, PDP-11, PDP-10, and PDP-8	Available only on CD-ROM

Table 8-2 lists the networking and installation supplements that have been archived.

Table 8-2 Archived Networking Manuals and Installation Supplements

Manual	Order Number
DECnet for OpenVMS Guide to Networking	AA-PV5ZA-TK
DECnet for OpenVMS Network Management Utilities	AA-PV61A-TK
DECnet for OpenVMS Networking Manual	AA-PV60A-TK
OpenVMS VAX Upgrade and Installation Supplement: VAX 8820, 8830, 8840	AA-PS6MA-TE
OpenVMS VAX Upgrade and Installation Supplement: VAX 8250, 8300, 8350	AA-PS6PA-TE
OpenVMS VAX Upgrade and Installation Supplement: VAX 8530, 8550, 8810 (8700), and 8820–N (8800)	AA-PS6QA-TE
OpenVMS VAX Upgrade and Installation Supplement: VAX 8600, 8650	AA-PS6UA-TE
TCP/IP Networking on OpenVMS Systems	AA-QJGDB-TE
VMS Upgrade and Installation Supplement: VAX-11/780, 785	AA-LB29B-TE
VMS Upgrade and Installation Supplement: VAX-11/750	AA-LB30B-TE

Descriptions of the archived OpenVMS manuals are as follows:

A Comparison of System Management on OpenVMS AXP and OpenVMS VAX

Discusses system management tools, the impact of Alpha page size on system management operations, the system directory structure, interoperability issues, and performance information. Designed for system managers who need to learn quickly how to manage an OpenVMS Alpha system.

Building Dependable Systems: The OpenVMS Approach

Offers practical information about analyzing the dependability requirements of your business applications and deciding how to use your computing systems to support your dependability goals. This information is complemented by technical summaries of the dependability features of OpenVMS and related hardware and layered software products.

COM, Registry, and Events for OpenVMS Developer's Manual

For programmers developing applications that move easily between the OpenVMS and Windows NT environments. Read this manual if you are encapsulating existing OpenVMS applications or data, or creating new COM applications for OpenVMS systems. It also provides information for those who want to use the OpenVMS Registry to store information about their OpenVMS systems alone, or who want to use the OpenVMS Registry as a shared repository for both OpenVMS and Windows NT registry information. This manual was formerly available online as the *OpenVMS Connectivity Developer's Guide*.

Creating an OpenVMS Alpha Device Driver from an OpenVMS VAX Device Driver

Describes the procedures for converting a device driver used on OpenVMS VAX to a device driver that runs on OpenVMS Alpha. This book also contains data structures, routines, and macros for maintaining an Alpha driver written in Macro-32.

Creating and OpenVMS AXP Step 2 Device Driver from a Step 1 Device Driver

Provides information for upgrading a Step 1 device driver (used in earlier versions of OpenVMS AXP) to a Step 2 device driver. A Step 2 device driver is required for OpenVMS AXP Version 6.1

Creating an OpenVMS AXP Step 2 Device Driver from an OpenVMS VAX Device Driver

Provides information for migrating a device driver used on OpenVMS VAX to a Step 2 device driver used on OpenVMS AXP Version 6.1

Guide to OpenVMS AXP Performance Management

Introduces and explains the techniques used to optimize performance on an OpenVMS Alpha system.

Guide to OpenVMS Performance Management

Introduces and explains the techniques used to optimize performance on an OpenVMS VAX system.

Migrating an Application from OpenVMS VAX to OpenVMS Alpha

Describes how to create an OpenVMS Alpha version of an OpenVMS VAX application. Provides an overview of the VAX to Alpha migration process and information to help you plan a migration. It discusses the decisions you must make in planning a migration and the ways to get the information you need to make those decisions. In addition, this manual describes the migration methods available so that you can estimate the amount of work required for each method and select the method best suited to a given application.

Migrating an Environment from OpenVMS VAX to OpenVMS Alpha

Describes how to migrate a computing environment from an OpenVMS VAX system to an OpenVMS Alpha system or a mixed-architecture cluster. Provides an overview of the VAX to Alpha migration process and describes the differences in system and network management on VAX and Alpha computers.

Migrating to an OpenVMS AXP System: Planning for Migration

Describes the general characteristics of RISC architectures, compares the Alpha architecture to the VAX architecture, and presents an overview of the migration process and a summary of migration tools provided by HP. The information in this manual is intended to help you define the optimal migration strategy for your application.

Migrating to an OpenVMS AXP System: Recompiling and Relinking Applications

Provides detailed technical information for programmers who must migrate high-level language applications to OpenVMS Alpha. Describes how to set up a development environment to facilitate the migration of applications, helps programmers identify application dependencies on elements of the VAX architecture, and introduces compiler features that help resolve these dependencies. Individual sections of this manual discuss specific application dependencies on VAX architectural features, data porting issues (such as alignment concerns), and the process of migrating VAX shareable images.

OpenVMS Alpha Guide to 64-Bit Addressing and VLM Features

Introduces and describes OpenVMS Alpha operating system support for 64-bit virtual addressing and Very Large Memory (VLM). Intended for system and application programmers, this guide highlights the features and benefits of OpenVMS Alpha 64-bit and VLM capabilities. It also describes how to use these features to enhance application programs to support 64-bit addresses and to efficiently harness very large physical memory.

OpenVMS Alpha System Dump Analyzer Utility Manual

Explains how to use the System Dump Analyzer utility to investigate system failures and examine a running OpenVMS Alpha system. Alpha programmers should refer to this manual; VAX programmers should refer to the *OpenVMS VAX System Dump Analyzer Utility Manual*.

OpenVMS AXP Device Support: Developer's Guide

Describes how to write a driver for OpenVMS Alpha for a device not supplied by HP.

OpenVMS AXP Device Support: Reference

Provides the reference material for the *Writing OpenVMS Alpha Device Drivers in C* by describing the data structures, macros, and routines used in device-driver programming.

OpenVMS Bad Block Locator Utility Manual

Describes how to use the Bad Block Locator utility to locate bad blocks on older types of media.

OpenVMS Compatibility Between VAX and Alpha

Compares and contrasts OpenVMS on VAX and Alpha computers, focusing on the features provided to users, system managers, and programmers.

OpenVMS Developer's Guide to VMSINSTAL

Describes the VMSINSTAL command procedure and provides guidelines for designing installation procedures that conform to standards recommended by HP. Intended for developers who are designing installation procedures for software products layered on the OpenVMS operating system.

OpenVMS DIGITAL Standard Runoff Reference Manual

Describes the DSR text-formatting utility.

OpenVMS EDT Reference Manual

Contains complete reference information for the EDT editor.

OpenVMS Exchange Utility Manual

Describes how to use the Exchange utility to transfer files between some foreign-format volumes and OpenVMS native volumes.

OpenVMS Glossary

Defines terms specific to OpenVMS that are used throughout the documentation.

OpenVMS Guide to Extended File Specifications

Provides an overview of Extended File Specifications and describes the overall differences and impact Extended File Specifications introduce to the OpenVMS environment.

OpenVMS Management Station Overview and Release Notes

Provides an overview and release notes for OpenVMS Management Station and describes how to get started using the software. OpenVMS Management Station is a powerful, Microsoft® Windows based management tool for system managers and others who perform user account and printer management tasks on OpenVMS systems. A revised version is available on the OpenVMS documentation CD-ROM and on the OpenVMS documentation web site.

OpenVMS Master Index

Offers an edited compilation of indexes from the manuals in the OpenVMS Full Documentation set.

OpenVMS National Character Set Utility Manual

Describes how to use the National Character Set utility to build NCS definition files.

OpenVMS Obsolete Features Manual

Presents the DCL commands, system services, RTL routines, and utilities made obsolete by VMS Version 4.0 through Version 5.0. Includes an appendix of DCL commands, RTL routines, and utilities eliminated from VMS Version 4.0.

OpenVMS Programming Environment Manual

Provides a general description of HP products and tools that define the programming environment. Introduces facilities and tools such as the compilers, the linker, the debugger, the System Dump Analyzer, system services, and routine libraries.

OpenVMS Programming Interfaces: Calling a System Routine

Describes the OpenVMS programming interface and defines the standard conventions to call an OpenVMS system routine from a user procedure. The Alpha and VAX data type implementations for various high-level languages are also presented in this manual.

OpenVMS RTL DECtalk (DTK$) Manual

Documents the DECtalk support routines contained in the DTK$ facility of the OpenVMS Run-Time Library.

OpenVMS RTL Parallel Processing (PPL$) Manual

Documents the parallel-processing routines contained in the PPL$ facility of the OpenVMS Run-Time Library. Indicates which routines are specific to Alpha or VAX, as well as how routines function differently on each system.

OpenVMS Software Overview

Provides an overview of the OpenVMS operating system and some of its available products.

OpenVMS SUMSLP Utility Manual

Describes how to use the SUMSLP batch-oriented editor to update source files.

OpenVMS System Messages and Recovery Procedures Reference Manual

Contains an alphabetical listing of the errors, warnings, and informational messages issued by the operating system. Also provides the meaning of each message and a statement of the action to be taken in response to each message. This manual is in two volumes.

OpenVMS Terminal Fallback Utility Manual

Describes how to use the Terminal Fallback utility to manage the libraries, character conversion tables, and terminal parameters that are available within this utility.

*OpenVMS VAX Card Reader, Line Printer, and LPA*11-K I/O User's REference Manual

Describes the card reader, laboratory peripheral accelerator, and line printer drivers on OpenVMS VAX.

OpenVMS VAX Device Support Manual

Describes how to write an OpenVMS VAX driver for a device not supplied by HP.

OpenVMS VAX Device Support Reference Manual

Provides reference material for the *OpenVMS VAX Device Support Manual* by describing the data structures, macros, and routines used in the device-driver programming.

OpenVMS VAX Patch Utility Manual

Describes how to use the Patch utility to examine and modify executable and shareable OpenVMS VAX images.

OpenVMS Wide Area Network I/O User's Reference Manual

Describes the DMC11/DMR11, DMP11, and DMF32, DR11-W, DRV11-WA, DR32, and asynchronous DDCMP interface drivers on OpenVMS VAX.

PDP-11 TECO User's Guide

Describes the operating procedures for the PDP-11 TECO (Text Editor and Corrector) program.

POLYCENTER Software Installation Utility User's Guide

Provides information on the POLYCENTER Software Installation utility, a new component that lets you install and manage software products that are compatible with the utility.

TCP/IP Networking on OpenVMS Systems

Provides an introductory overview of TCP/IP networking and describes OpenVMS DCL support for TCP/IP capabilities.

Symbols

$GETRMI system service, 66

Numerics

3D direct rendering, 73
64-bit pointer support, 54

A

ACME
  agent, 53
  SDK, 53
AlphaServer Series systems
  DS15, 34
  ES47, ES80, GS1280, 34
ANALYZE/DISK/SHADOW command, 51
ATI RADEON 7500 Graphics, 71

B

Backup utility
  command qualifiers, 34
BALSETCNT system parameter, 50
Blocks-to-bytes display, 25
Buffer size, 66

C

C RTL, 54
  child process, 54
  features logicals, 55
  performance enhancements, 55
  POSIX style indentifiers, 54
CDSA, 44, 73
Child process, 54
Clusters, 42, 47
Commands
  ANALYZE/DISK/SHADOW, 51
  COPY, 26
  SDA, 67
  SEARCH, 26
  SET DEVICE/RESET, 49
  SET SHADOW, 51
  SHOW SHADOW, 51
Common Data Security Architecture, *See* CDSA
Component indictment, 37
Configuration procedures, 44
COPY command, 26

D

Data compression, 48
DCL
  command functions, 25
  command-size limits, 27
DDT intercept establisher, 57
DECdtm, 65
DEGXA adapter, 44, 66
Dissimilar device shadowing (DDS), 39
DS15 system support, 34
DUMP_PRIORITY commands
  SYSMAN, 49
DVI$_SHDW_SITE
  $GETDVI item code, 66
Dynamic volume expansion (DVE), 39

E

ELV, 40
Error Log Viewer, *See* ELV
Euro symbol, 40
Extended DCL (EDCL), 27
Extended File Cache, 41

F

failSAFE IP, 45
Fast Path, 42, 43, 47
Features logicals, 55
Fibre Channel arbitrated loop configuration, 43
FIBRE_SCAN Utility, 43
File and module generation numbers, 49
Font support, 40

G

Generation numbers
  file, 49
  module, 49
Gigabit Ethernet adapter, 44, 66

H

HP StorageWorks modular SAN array 1000, 43

I

Installation procedures, 44
IOC_STD$DEVCONFIG_DEREGISTER routine, 64
IOC_STD$DEVCONFIG_REGISTER routine, 62
IOC_STD$ESTABLISHER_DDT_ALTSTART routine, 59
IOC_STD$ESTABLISHER_DDT_CANCEL routine, 60
IOC_STD$ESTABLISHER_DDT_MNTVER routine, 61
IOC_STD$ESTABLISHER_DDT_START routine, 59
IOLOCK8 spinlock, 42, 47
iType font rasterizer, 40

J

Jumbo receive buffers, 66

K

Kerberos, 44, 75
KZPDC adapter, 43

L

LAN, 42, 47
  adapters, 42, 47
  drivers, 66
  Failover, 45
  Fast Path, 42
  portlock, 47
Large pool lookaside lists, 66

M

Mail utility, 46
Message reduction, 46
MIME utility, 31
Monitor support, 73
Mount verification modifications, 46
MSA1000, 43
Multipath for SCSI and Fibre Channel, 57
Multiple RADEON cards, 72
MVSUPMSG_INTVL system parameter, 46
MVSUPMSG_NUM system parameter, 46

O

OCLA utility, 67
OpenVMS e-Business Infrastructure Package, 74

P

Patch removal, 47
PEdriver, 47
PKRdriver, 43
POLYCENTER Software Installation utility
  running in recovery mode, 47
  uninstalling a patch, 47
POSIX style identifiers, 54
Procedures
  configuration, 44
  time zone, 44
  upgrade, 44

R

RADEON 7500 graphics, 71
RAID controller, 43
Recovery mode, 47
Routines
  IOC_STD$DEVCONFIG_DEREGISTER, 64
  IOC_STD$DEVCONFIG_REGISTER, 62
  IOC_STD$ESTABLISHER_DDT_ALTSTART, 59
  IOC_STD$ESTABLISHER_DDT_CANCEL, 60
  IOC_STD$ESTABLISHER_DDT_MNTVER, 61
  IOC_STD$ESTABLISHER_DDT_START, 59

S

SCS/IOLOCK, 47
SDA commands, 67
SEARCH command, 26
Secure Sockets Layer, *See* SSL
Security, 75
Set default directory, 54
SET DEVICE command, 49
SET SHADOW command, 51
SETSHOSHADOW utility, 51
SHADOW_REC_DLY system parameter, 50
SHADOW_SITE_ID system parameter, 50
SHOW SHADOW command, 51
Smart Array 53.xx, 43
SMP, 42, 47
Spinlock, 42, 47
SSL, 76
Symmetric multiprocessing, *See* SMP
System Dump Analyzer, 67
System Event Analyzer (SEA) utility, 37
System Management utility (SYSMAN) commands
  DUMP_PRIORITY, 49
System parameters
  BALSETCNT, 50
  MVSUPMSG__INTVL, 46
  MVSUPMSG__NUM, 46
  SHADOW_REC_DLY, 50
  SHADOW_SITE_ID, 50
  WSMAX, 50

T

Third-party product compatiblity, 57
Time zone procedure, 44

U

Ultra3 SCSI, 43
UNIX portability, 18, 50
Upgrade procedures, 44
USB Configuration Manager (UCM) utility, 51
Utilities
  ANALYZE/DISK, 51
  Backup, 34
  Error Log Viewer (ELV), 40
  FIBRE_SCAN, 43
  Mail, 46
  MIME, 31
  OCLA, 67
  POLYCENTER Software Installation, 47
  SETSHOSHADOW, 51
  System Event Analyzer (SEA), 37
  System Management (SYSMAN), 49
  USB Configuration Manager (UCM), 51

V

Video modes supported, 72
Volume Shadowing for OpenVMS, 39
  management functions, 51

W

WSMAX system parameter, 50

X

XFC, 41