The OpenVMS Frequently Asked Questions(FAQ)

ABSTRACT

This document contains answers to many common questions on the OpenVMS operating system, and covers OpenVMS, and OpenVMS running on Itanium, Alpha, and VAX hardware.

Revision/Update Information   OpenVMS V7.3-1

Revision/Update Information: Revision Date/June 2002

FAQ

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

Contents

xviii

Contents

Contents

xx

Contents

INDEX

TABLES

Preface

OpenVMS FAQ

This is the OpenVMS Frequently Asked Questions (FAQ) posting for the comp.os.vms and comp.sys.dec usenet newsgroups.

The OpenVMS FAQ is archived in (at least) the following locations:

- http://www.openvms.compaq.com/
- ftp://rtfm.mit.edu/pub/usenet/comp.os.vms/
- comp.answers and news.answers newsgroups
- http://www.faqs.org/
- http://www.google.com/
- http://www.kjsl.com/vmsfaq
- http://eisner.decus.org/vms/faq.htm

Other internet FAQs are generally available in these locations:

- comp.answers and news.answers newsgroups
- ftp://rtfm.mit.edu/pub/usenet/...
- http://www.faqs.org/
- http://www.google.com/

Note

Please send suggestions for changes or for additions to the FAQ to the FAQ Editor:

Please do *not* send technical questions to the FAQ Editor. The FAQ editor is *not* in a position to answer general questions, nor to provide support. Rather, please post your questions to the appropriate newsgroup, or please contact your support organization. Please also consider reading Section 1.3 before posting to the newsgroups.

Your understanding in this is greatly appreciated.

Contributors

Many people have contributed to this FAQ document, directly and indirectly. In many cases, part or all of an answer has been adapted from one or more postings from the comp.os.vms or other newsgroup. (The FAQ editor wishes to thank to all of those folks who post answers to the newsgroups, and that have contributed to the contents of this FAQ.)

The following is an incomplete list of the folks that have contributed directly or indirectly to the contents of the OpenVMS FAQ. Additions, corrections, and updates to this list of contributors and to the contents of the FAQ itself are welcome, and all omissions and misspellings are unintentional. Thanks to each of the contributors here, and to all of the folks that have participated in the FAQ and in the newsgroups.

Paul Anderson, Jason Armistead, Zoltan Arpadffy, John AtoZ, Bailey, Jeremy Begg, Colin Blake, Ken Blaylock, Bol, Craig Berry, Mark Berryman, Jim Becker, Verell Boaen, Bol, Richard Brodie, Robert Alan Byer, Jeff Campbell, Antonio Carlini, Ken Chaney, Cristy, John Croll, David J. Dachtera, Robert Deininger, Dale Dellutri, Leo Demers, Sue Denham, Thomas Dickey, Eric Dittman, Jim Dunham, Eric, Glenn Everhart, Ken Fairfield, Yaacov Fenster, Gunther Froehlin, Harry Garonzik, Derek Garson, Clair Grant, Hunter Goatley, Ruth Goldenberg, Vance Haemmerle, Ki Suk Hahn, SHarris, Fletcher Hearns, Phillip Helbig, Stephen Hoffman, Horn, Jaf, Leif Jansson, Kevin Jenkins, Henry Juengst, inazu_k, Geoff Kingsmill, Fred Kleinsorge, Felix Kreisel, Veli Korkko, Ruslan R. Laishev, Norm Lastovica, p_lee, Jerry Leichter, Richard Levitte, Steve Lionel, Kerry Main, John E. Malmberg, David Mathog, Dale Miller, Dick Munroe, Patrick Moreau, Morrison, otis, Keith Parris, Plass, pvhp, David Rabahy, Stan Rabinowitz, Mike Raspuzzi, Steve Reece, Aaron Sakovich, Brian Schenkenberger, Atlant Schmidt, Wayne Sewell, Steven Shamlian, Scott Snadow, Scott Squires, stfp, Dave Sweeney, Mike Thompson, Arne Vajhøj, Martin Vorländer, Peter Weaver, William Webb, Williams, Paul Winalski, Uwe Zinser.

Again, thanks to each of these individuals for their assistance and for their contributions, and particularly for sharing their expertise.

The Fine Print

All trademarks, all registered trademarks, and all servicemarks used or referenced here are the properties of their respective owner(s) or holders.

The redistribution, reposting and/or the web-serving of complete and unaltered copies of this FAQ document is permitted. Other uses only with prior permission.

While the information contained here is believed correct, the use of any and all information acquired from this document is entirely at the risk of the user(s).

Though the editor of this FAQ is an employee of Hewlett-Packard Company (HP), this posting and the information contained within are not official statements of HP.

1 Introduction

1.1 What core OpenVMS URLs do you need to know?

If you have no other OpenVMS URLs bookmarked, you will want to be familiar with the URLs listed in Table 1–1:

Table 1–1 Core Websites

URL	Contents
http://www.openvms.compaq.com/	
	Core OpenVMS Information
http://www.openvms.compaq.com/openvms/roadmap/openvms_roadmaps.htm	
	OpenVMS Future Release Contents, Schedules
http://www.openvms.compaq.com/doc/ http://www.openvms.compaq.com:8000/ http://www.openvms.compaq.com/commercial/	
	OpenVMS and Core Layered Product Documentation
http://askq.compaq.com/ http://ftp.digital.com.au/pub/ecoinfo/ ftp://ftp.service.digital.com/public/vms/vax/... ftp://ftp.service.digital.com/public/vms/axp/...	
	Core OpenVMS Support Search Engine URLs, FTP Patch Area
http://www.openvms.compaq.com/freeware/	
	The OpenVMS Freeware

1.2 What are the OpenVMS Usenet Newsgroups?

There are a huge number of newsgroups available, some of which discuss topics of interest to folks using OpenVMS operating system. Key among these are the comp.os.vms newsgroup and the vmsnet.* heirarchy of newsgroups.

1.2.1 What is the scope of the comp.os.vms newsgroup?

The comp.os.vms newsgroup is the primary newsgroup for discussion of the HP OpenVMS operating system and the computer systems on which it runs. Questions about layered products which run on OpenVMS are also welcome, though many of them (in particular, language compilers and database systems) have more specific newsgroups. If a question has some relationship to OpenVMS, it belongs in comp.os.vms.

1.2.2 What newsgroups carry VMS-related information?

The vmsnet.* hierarchy is operated by the Encompass users group (formerly known as DECUS), and contains various newsgroups of interest, including vmsnet.misc and vmsnet.alpha.

Sprechen Sie Deutsch? Besuchen Sie bitte den newsgroup de.comp.os.vms.

The comp.sys.dec newsgroup carries general discussions about various HP computer systems, and specifically systems that were acquired by HP as part of the acquisition of Compaq, which had previously been acquired by Compaq as part of the acquisition of Digital Equipment Corporation (DIGITAL).

Also of interest are resources and conferencing systems including Encompasserve (formerly known as DECUServe), the OpenVMS.Org forums, and the Compaq Working Group folks:

- telnet://eisner.decus.org/
- http://www.openvms.org/
- http://www.compaqworkinggroup.org/

1.2.3 What newsgroup archives are available?

Extensive archives of INFO-VAX mailings and comp.os.vms postings are archived and available at ftp://crvax.sri.com/info-vax/. Google also has extensive newsgroup arechives available at http://www.google.com/.

1.2.4 What is INFO-VAX?

INFO-VAX is a mailing list which is bidirectionally gatewayed to the comp.os.vms newsgroup. This means that postings to comp.os.vms get automatically sent to INFO-VAX subscribers and messages sent to the INFO-VAX list are automatically posted to comp.os.vms.

Please remember that propogation delays can and do vary, meaning that mailings and postings may be delayed or even lost. It is quite possible that postings may not be delivered for several days, and that some postings will appear out of the expected order.

For details on the available archives, please see Section 1.2.3.

1.2.4.1 How do I subscribe to or unsubscribe from INFO-VAX?

The address for subscription requests, as well as notes intended for the moderator, is Info-VAX-Request@Mvb.Saic.Com. Subscription requests are handled automatically by a mail server. This mail server ignores the subject line and processes each line of the message as a command. The syntax for subscribing and unsubscribing and setting digest or non-digest modes is:

```
SUBSCRIBE INFO-VAX      (ADD is a valid synonym)
UNSUBSCRIBE INFO-VAX (REMOVE, SIGNOFF, and SIGN-OFF are valid synonyms)
SET INFO-VAX DIGEST     (to receive in Digest format)
SET INFO-VAX NODIGEST   (to receive each message individually)
```

The mail server is not case-sensitive; case is irrelevant. Attempts to fetch copies of the mailing list are rejected for reasons of confidentiality. Any message not understood by the mailserver will be forwarded to a human (allegedly) for manual processing.

1.3 What is etiquette?

Before posting or emailing a question, please use the available local resources, such as the OpenVMS manuals, the HELP, and the resources and information in and referenced by this FAQ. Please use these first. Also please specifically read the release notes and (if appropriate) the cover letter for the product you are using. (The release notes are generally placed in SYS$HELP:.) Quite often, these simple steps will allow you to quickly find the answer to your own question—and more quickly than waiting for a response to question posted to a newsgroup, too. These steps will save you time, and will also help ensure you have a good reputation with the folks that might be included to answer one of your

future questions, a question not covered in these resources. Put another way, if you do not want your questions to be ignored in the future—and please remember that the folks in the newsgroups do *not* have to answer your questions—you won't want to "annoy the natives" by asking a question that has already been answered far more times more than you might have realized, or a question whose answer is readily available had you made a small effort.

When posting, please consider the following suggestions:

- Include a valid e-mail address in the text of your posting or in a "signature" appended to the end. Reply-to addresses in headers often get garbled. Anonymous addresses can also simply be ignored, as fake addresses are regularly used by folks that are "trolling" and by folks that are spamming.

- If you are submitting a question, please be as specific as you can. Include relevant information such as processor type, product versions (OpenVMS and layered products that apply), error message(s), DCL command(s) used, and a short, reproducible example of problems. Say what you've tried so far, so that effort isn't duplicated. Keep in mind that there's not yet a telepathy protocol for the Internet. (The more detailed your description, the better that people can help you with your question.)

- If responding to a posting, include in your reply only as much of the original posting as is necessary to establish context. As a guideline, consider that if you've included more text than you've added, you've possibly included too much. Never include signatures and other irrelevant material.

- Please be polite. If the question isn't worded the way you think is correct or doesn't include the information you want, try to imagine what the problem might be if viewed from the poster's perspective. Requests for additional information are often better sent through mail rather than posted to the newsgroup.

- If you have a problem with HP (or any other vendor's) product, please use the appropriate support channel. Don't assume that newsgroup postings will get read, will be responded to by the appropriate developers, or will be later followed up on...

- If you are posting from a web browser, news reader or if you are posting via email sent to INFO-VAX, please turn off MIME, vcard, attachments, and other mechanisms that assume anyone reading the post has the corresponding capability—use the text-only option of

your web browser, news reader, or mailer. Usenet is traditionally a text-only medium, and many comp.os.vms participants will use tools that have this support disabled, or that do not have this support. If the message uses MIME or attachments or such, the text of your message will be buried in a large pile of gibberish, and some tools will send multiple copies of the text within a single posting.

Before posting your question to the comp.os.vms newsgroup or sending your message to the INFO-VAX list, also please take the time to review available etiquette information, such as that included in the following documents:

- ftp://rtfm.mit.edu/pub/usenet/news.answers/usenet/primer/part1
- ftp://rtfm.mit.edu/pub/usenet/news.answers/usenet/faq/part1
- ftp://rtfm.mit.edu/pub/usenet/news.answers/usenet/emily-postnews/part1
- ftp://rtfm.mit.edu/pub/usenet/news.answers/usenet/writing-style/part1
- ftp://rtfm.mit.edu/pub/usenet/news.answers/usenet/posting-rules/part1

This information will document the etiquette of newsgroups, as well as providing you with the knowledge the vast amount of newsgroup-related information that is readily available to you, and where to find it...

Note

Please do not post security holes or system crashers

Rather, please report these problems directly to HP. Why? So that HP has a change to resolve and distribute a fix before other customer sites can be affected. Most folks in the newsgroups are honest and deserve to know about potential security problems, but a few folks can and will make nefarious use of this same information. Other sites will hopefully return the courtesy, and not post information that will potentially compromise your site and your computer environment.

1.4   What OpenVMS user group(s) are available?

Encompass, the Enterprise Computing Association, is a user group comprised of information technology professionals that are interested in the Enterprise-oriented products, services, and technologies of Compaq and of the former DIGITAL. Encompass offers newsletters, the Encompass website, and offers various gatherings and related services, including symposia events and local users group meetings.

Encompass is a descendent of the organization known as DECUS, the Digital Equipment Computer Users Society.

For more information on Encompass, please visit the Encompass web site:

- http://www.encompassus.org/

The organization comprised of customers of Hewlett-Packard Company (HP) that is probably most analogous to the Encompass organization is Interex:

- http://www.interex.org/

Like Encompass, Interex offers various services and events of interest to folks that presently work with and/or that wish to learn about HP products and offerings. Please see the Interex website for details.

1.5   OpenVMS Support, Questions and Comments?

The following section includes contacts for OpenVMS Feedback, and information on how to obtain technical support information.

1.5.1   Corporate contacts for OpenVMS Business Issues?

The following folks are the HP corporate contacts for OpenVMS business issues, listed in descending order from Carlton Fiorina, President of Hewlett-Packard Company to Mark Gorham, the Vice President of the OpenVMS Engineering organization:

- Carlton Fiorina
- Peter Blackmore
- Scott Stallard
- Richard Marcello
- Mark Gorham

These folks will quite obviously respond best to cogently-worded OpenVMS corporate-level business issues. These folks are NOT appropriate contacts for any OpenVMS technical support issues nor for any OpenVMS technical support requests, nor for any other non-corporate-related, non-business-related issues.

1.5.2 Contact for OpenVMS Marketing Issues and Questions?

Please see Section 3.5.

1.5.3 Contact URLs for OpenVMS Technical Issues?

For technical issues and technical support, please contact your software support organization, or your local HP Customer Support Center or HP Reseller. In North America, you can call 1-800-OK-COMPAQ.

Please remember to review the following useful support URLs:

- http://askq.compaq.com/
- http://www.openvms.compaq.com/wizard/
- ftp://ftp.service.digital.com/public/vms/vax/...
- ftp://ftp.service.digital.com/public/vms/axp/...

2 General Information

2.1 What is OpenVMS? What is its history?

OpenVMS, originally called VMS (Virtual Memory System), was first conceived in 1976 as a new operating system for the then-new, 32-bit, virtual memory line of computers, eventually named VAX (Virtual Address eXtension).

The first VAX model, the 11/780, was code-named "Star", hence the code name for the VMS operating system, "Starlet", a name that remains to this day the name for the system library files (STARLET.OLB, etc.).

VMS version X0.5 was the first released to customers, in support of the hardware beta test of the VAX-11/780, in 1977. VAX/VMS Version V1.0 shipped in 1978, along with the first revenue-ship 11/780s.

OpenVMS was designed entirely within HP and specifically within the former Digital Equipment Corporation (DIGITAL). Two of the principal designers were Dave Cutler and Dick Hustvedt, though with a wide variety of other contributors. OpenVMS was conceived as a 32-bit, virtual memory successor to the RSX-11M operating system for the PDP-11. Many of the original designers and programmers of OpenVMS had worked previously on RSX-11M, and many concepts from RSX-11M were carried over to OpenVMS.

OpenVMS VAX is a 32-bit, multitasking, multiprocessing virtual memory operating system. Current implementations run on VAX systems from HP and other vendors.

OpenVMS Alpha is a 64-bit multitasking, multiprocessing virtual memory operating system. Current implementations run on Alpha systems from HP, and other vendors.

Work to port OpenVMS to systems based on the Intel IA-64 architecture and specifically to the Itanium Processor Family is presently underway.

For more details on OpenVMS and its features, please read the OpenVMS Software Product Description at:

- http://www.compaq.com/info/spd/
  OpenVMS typically uses SPD 25.01.xx and/or SPD 41.87.xx.

Additional information on the general features of various OpenVMS releases, release dates, as well as the development project code names of specific releases, is available at:

- http://www.openvms.compaq.com/openvms/os/openvms-release-history.html

Additional historical information—as well as pictures and a variety of other trivia—is available in the VAX 20th anniversary book:

- http://www.openvms.compaq.com/openvms/20th/vmsbook.pdf

For information on the FreeVMS project, and on hobbyist and educational versions of OpenVMS, please see:

- http://www.free-vms.org/

- http://www.montagar.com/hobbyist/

- http://www.openvmsedu.com/

Also please see the related software licensing topics Section 2.7 and Section 2.14.

2.2   What is the difference between VMS and OpenVMS?

VMS and OpenVMS are two names for the same operating system. Originally, the operating system was called VAX-11/VMS; it changed to VAX/VMS at around VAX/VMS V2.0. When the VMS operating system was ported to the Alpha platform, it was renamed OpenVMS, for both VAX and Alpha (and for the Itanium Processor Family), in part to signify the high degree of support for industry standards such as POSIX, which provides many features of UNIX systems.

For those versions with POSIX, an OpenVMS license allows you to install and run POSIX for OpenVMS at no additional charge; all you need is the media and documentation which can be found on the Consolidated Distribution and On-Line Documentation CD-ROMs. Support for the POSIX package on more recent OpenVMS releases is not available, various parts of POSIX such as calls from the API are being integrated more directly into OpenVMS. For more information on POSIX for VMS see question SOFT2

What became confusing is that the OpenVMS name was introduced first for OpenVMS AXP V1.0 causing the widespread misimpression that OpenVMS was for Alpha AXP only, while "regular VMS" was for VAX. In fact, the official name of the VAX operating system was changed as of

V5.5, though the name did not start to be actually used in the product until V6.0.

The proper names for OpenVMS on the two platforms are now "OpenVMS VAX" and "OpenVMS Alpha", the latter having superseded "OpenVMS AXP".

2.2.1   How do I port from VMS to OpenVMS?

You already did. Wasn't that easy? Please see Section 2.2 for details.

2.3   Which is better, OpenVMS or UNIX?

This question comes up periodically, usually asked by new subscribers amd new posters who are long-time UNIX or Linux users. Sometimes, the question is ignored totally; other times, it leads to a long series of repetitive messages that convince no one and usually carry little if any new information. Please do everyone a favor and avoid re-starting this perpetual, fruitless debate.

That said, OpenVMS and the better implementations of UNIX are all fine operating systems, each with its strengths and weaknesses. If you're in a position where you need to choose, select the one that best fits your own requirements, considering, for example, whether or not the layered products or specific OS features you want are available.

2.4   Is HP continuing funding and support for OpenVMS?

Yes.

Active development of new OpenVMS releases is underway, as well as the continuation of support.

Please see the following URLs for details, roadmaps, and related information:

- http://www.openvms.compaq.com/
- http://www.openvms.compaq.com/OPENVMS/strategy.html
- http://www.openvms.compaq.com/openvms/roadmap/openvms_roadmaps.htm
- http://www.openvms.compaq.com/openvmstimes/
- http://www.compaq.com/inform/

2.5 What OpenVMS CD-ROM distribution kits are available?

Various distributions are available.

For information on the available part numbers and current products (OpenVMS distribution kits, media, documentation, etc) and associated licensing information, please see the OpenVMS Software Product Description (SPD), available at:

- http://www.compaq.com/info/spd/
  OpenVMS typically uses SPD 25.01.xx and/or SPD 41.87.xx.

The CD-ROMs listed in Table 2–1 contain just the OpenVMS Alpha operating system. These are bootable, and can be used to run BACKUP from CD-ROM.

Table 2–1   OpenVMS Media Kits

Part	Description
QA-MT1AP-H8	OpenVMS Alpha V6.1-1H2 hardware release CD-ROM
QA-MT1AG-H8	OpenVMS Alpha V6.2-1H3 hardware release CD-ROM
QA-MT1AD-H8	OpenVMS Alpha V7.1-1H1 hardware release CD-ROM
QA-MT1AR-H8	OpenVMS Alpha V7.1-2 maintenance release CD-ROM
QA-MT1AT-H8	OpenVMS Alpha V7.2-1 maintenance release CD-ROM
QA-MT1AU-H8	OpenVMS Alpha V7.2-1H1 hardware release CD-ROM

The table Table 2–2 contains the consolidated ECO distribution kit subscriptions, and these provide sites with eight updates of the current ECO kits per year:

Table 2–2   OpenVMS ECO Kits

Part	Description
QT-3CQAA-C8	OpenVMS Alpha
QT-3CRAA-C8	OpenVMS VAX

The OpenVMS VAX and OpenVMS Alpha source listings CD-ROM sets listed in Table 2–3 include the source listings of most of OpenVMS, and these CD-ROM sets are invaluable for any folks working directly with

OpenVMS internals, as well as folks interested in seeing examples of various programming interfaces.

Table 2–3   OpenVMS Source Listings CD-ROM Kits

Part	Description
QB-MT1AB-E8	OpenVMS Alpha Source Listings CD-ROM
QT-MT1AB-Q8	OpenVMS Alpha Source Listings CD-ROM Updates
QB-001AB-E8	OpenVMS VAX Source Listings CD-ROM
QT-001AB-Q8	OpenVMS VAX Source Listings CD-ROM Updates

2.6   In what language is OpenVMS written?

OpenVMS is written in a wide variety of languages.

In no particular order, OpenVMS components are implemented using Bliss, Macro, Ada, PLI, VAX and DEC C, Fortran, UIL, VAX and Alpha SDL, Pascal, MDL, DEC C++, DCL, Message, and Document. And this is certainly not a complete list. However, the rumor is NOT true that an attempt was made to write pieces of OpenVMS in every supported language so that the Run-Time Libraries could not be unbundled. (APL, BASIC, COBOL and RPG are just some of the languages NOT represented!)

There are a large variety of small and not-so-small tools and DCL command procedures that are used as part of the OpenVMS build, and a source code control system capable of maintaining over a hundred thousand source files across multiple parallel development projects, and overlapping releases.

2.7   How do I obtain or transfer a VMS license?

If you are an Encompass member and are considering acquiring and using a VAX or Alpha system for hobbyist (non-commercial) use, (free) licenses for OpenVMS VAX and OpenVMS Alpha are available to Encompass members. In addition to the license, VAX and Alpha distribution CD-ROM kits are available with OpenVMS, DECwindows Motif, DECnet and TCP/IP networking, compilers, and a variety of layered products.

For further information, link to:

- http://www.montagar.com/hobbyist/

On the hobbyist license registration form (as of August 2001), you are offered the choice of the "OpenVMS VAX" license(s), the "OpenVMS Alpha" license(s), and the "Layered Products" licenses. You will want the operating system license for your particular OpenVMS platform *and* you will want the "Layered Products" licenses.

For vendors wishing to license products specifically for hobbyist use (and to not issue hobbyist PAKs), the program provides hobbyists with the license PAK OPENVMS-HOBBYIST.

For information on OpenVMS licenses for educational customers, please see the HP Campus Software License Grant (CSLG) license program and the OpenVMS Educational license program:

- http://www.openvmsedu.com/

To transfer a commercial OpenVMS license from one owner to another, or to purchase a commercial license, you can contact HP at 1-800-OK-COMPAQ (in North America), or your local or regional sales office or reseller.

Commercial developers can join the HP Solutions Alliance (CSA) program, and can (potentially) receive discounts on various software product licenses and software distributions, as well as on hardware purchases. Please see Section 2.14.

2.8 Does OpenVMS support the Euro currency symbol?

OpenVMS can generate the *%xA4* character code used for the Euro, and the DECwindows DECterm can display the glyph. Please check with the vendor of your terminal or terminal emulator for additional details.

For additional information on the support of the European Monetary Union Euro currency glyph on OpenVMS, please see:

- http://www.openvms.compaq.com/euro/

2.9 Why hasn't OpenVMS been ported to Intel (IA-32) systems?

Why? Business reasons...

There is presently a belief that there would be insufficient market to justify the cost involved in porting OpenVMS to systems using the Intel IA-32 architecture. In addition to the direct costs involved in any port, each maintainer of a product or a package for OpenVMS has to justify the

port to "OpenVMS Pentium" or to OpenVMS Itanium, akin to the required justifications for a product port from OpenVMS VAX to OpenVMS Alpha.

But yes, it would be nice to have.

And yes, OpenVMS Engineering is well aware of AMD Opteron/Hammer, too.

2.10 Are there any network-accessible OpenVMS systems?

Yes, though various restrictions can and do apply.

- Hobbes
  Hobbes is a MicroVAX 3100 Model 40 for which free access and accounts are available to OpenVMS enthusiasts. This system has BASIC, Pascal, Fortran, and C compilers installed. If you would like an account on Hobbes, please see the FAQ at http://www.hobbesthevax.com/.

- OpenVMS Galaxy Test Drive
  HP currently offers an OpenVMS Galaxy Test Drive system, based on an AlphaServer 4100 series configured as two instances of the OpenVMS operating system. For details, please see http://www.testdrive.compaq.com/galaxy/

- HP CSA Test Drive
  http://www.testdrive.compaq.com/

- Encompasserve
  telnet://eisner.decus.org/

2.11 What version of OpenVMS do I need?

For information on supported platforms, please see the OpenVMS Software Product Description (SPD) for the particular OpenVMS version of interest.

- http://www.compaq.com/info/spd/
  OpenVMS typically uses SPD 25.01.xx and/or SPD 41.87.xx.

For a table of OpenVMS versions for various platforms, please see:

- http://www.openvms.compaq.com/openvms/supportchart.html

For information on the Multia, related Alpha single-board computers, or other officially unsupported systems, please see Section 14.5 and Section 14.10.

The following is a rule-of-thumb for Alpha platform support. The table
Table 2–4 contains the earliest OpenVMS Alpha release with support for
a particular series of Alpha microprocessors:

Table 2–4   OpenVMS Alpha Version Rule-Of-Thumb

Microprocessor Generation	OpenVMS Version	General Comments
21064 EV4	V1.0	few systems; most EV4 require later; upgrade available
21164 EV5	V6.2	subsequent upgrade available
21164A EV56	V6.2-1H3	subsequent upgrade to V7.1 and later
21264 EV6	V7.1-2	subsequent upgrade typically to V7.2-1 or later
21264A EV67	V7.1-2	subsequent upgrade typically to V7.2-1 or later
xxxxxx EV68	V7.2-1	believed/probable; currently expectation

2.12 How can I submit OpenVMS Freeware?

For the guidelines and submission info, please visit the URL:

• http://www.openvms.compaq.com/openvms/freeware/cd_guide.html

To order the current OpenVMS Freeware CD-ROM kit (shipping and handling charges apply), please request part number QA-6KZAA-H8.

2.13 Porting applications to OpenVMS?

Porting can range from simple to rather complex, and depends on the features used on the original platform.

This section covers generic porting, and porting from OpenVMS VAX to OpenVMS Alpha. (Porting from OpenVMS VAX to OpenVMS Alpha is often quite simple and involves little more than rebuilding from source, though a few applications using features specific to VAX will require some additional effort to port.)

Several manuals on porting from OpenVMS VAX to OpenVMS Alpha are available in the OpenVMS documentation set, including information on porting VAX Macro32 assembler code to the Macro32 compiler on OpenVMS Alpha, on management differences, on upgrading privileged code, and application migration:

• http://www.openvms.compaq.com:8000/

• http://www.openvms.compaq.com/doc/

• http://www.openvms.compaq.com/commercial/

Details on the C programming environment are available at:

• http://www.openvms.compaq.com/commercial/c/c_index.html

Details on porting VAX C to HP C are are available at:

• http://www.openvms.compaq.com/commercial/c/index_vax.htm

An OpenVMS Porting Library is available at:

• http://www.openvms.compaq.com/ebusiness/Technology.html

Information on the Enterprise Toolkit, a Visual-based development environment for developing applications for OpenVMS using a Microsoft platform, is available at:

- http://www.openvms.compaq.com/commercial/et/et_index.html

Details on DCE, CORBA, BridgeWorks, and COM/DCOM middleware is available at:

- http://www.openvms.compaq.com/commercial/middleware.html

Information on the COE standards is available at:

- http://diicoe.disa.mil/coe/

A wide variety of programming development tools and middleware are available as commercial products (eg: DECset, IBM MQseries), and various tools are also available as shareware or Freeware. Please see other sections of this FAQ, and please see:

- http://www.compaq.com/csa/directory/

2.14   How do I join HP Solutions Alliance?

The HP Solutions Alliance (CSA) is a (free) program that is open to and that supports software partners, consultants, and service providers:

- http://www.compaq.com/csa/

CSA provides members with discounts on hardware, porting assistance, and many other benefits.

For those familiar with the program, the DIGITAL Association of Software and Application Parterns (ASAP) program has been incorporated into CSA.

The HP Solutions Alliance Technical Journal (CTJ) is "web-published" monthly, and available at:

- http://csa.compaq.com/CompaqTechnicalJournal.html

2.15 memory management, resource management, process scheduling, etc?

So you have been instructed to write a school research paper on OpenVMS, and you need technical content on the OpenVMS Virtual Memory System, on any memory segmentation, on OpenVMS Resource Management, on the OpenVMS File System, on the OpenVMS user interface, etc.

Invariably, your professor/instructor/teacher will ask you a series of questions. Most commonly, the questions will request descriptions of one or more of the following items, and at varying levels of detail:

- process scheduling algorithm(s)
- Interprocess comunications
- Process or system synchronization constructs
- Memory management and/or virtual memory implementation
- RMS or XQP file structures
- Resource management
- History of HP OpenVMS
- History of Compaq and/or of Digital Equipment Corporation (DEC)

Any particular presentation or research paper, and particularly a scholastic presentation, can have many different potential target audiences, and very different presentation levels. Further, the usual underlying reason for scholastic presentations and scholastic research projects really has little to do with the subject matter, it is a task specifically intended to teach the student(s) (eg: you) how to perform the research. The instructor already knows most of (all of?) the information that you have been asked to collect.

For very technical details on OpenVMS and OpenVMS internals, the book you want is the Internals and Data Structures Manual (IDSM), available in your school or computing center library, and the IDSM can also be purchased. Additional technical details of the Alpha microprocessor are available in the Alpha Architecture Reference Manual documentation that is available for download. (Pointers to Alpha technical documentation are available in Section 14.7, and elsewhere.)

For higher-level (less technical) details, the OpenVMS documentation set is available on-line. The Programming Concepts and the File Systems manual are probably the best manuals to start with, depending on the particular level of detail the research requires.

And please understand the hesitation of various folks to provide you with a completely-written research report on your topic. Why? We might have to work with you after you graduate—you need to know how to perform at least basic research on your own, regardless of the topic.

3 Documentation

3.1 Where can I find online copies of OpenVMS manuals?

The HP OpenVMS and HP Layered Product documentation is copyrighted material.

HTML format on-line product documentation sets for specific HP OpenVMS products are presently available at:

- http://www.openvms.compaq.com:8000/
- http://www.openvms.compaq.com/doc/
- http://www.openvms.compaq.com/commercial/

Documentation is offered on separately orderable CD-ROM media through a subscription to the Consolidated On-Line Documentation (ConOLD) product (see Section 2.5.) ConOLD manuals are readable with BNU, a viewer that is supplied with the documentation distribution. BNU can display HTML, Bookreader, and documentation in other formats.

MGBOOK, a viewer for Bookreader-format documentation is available for character-cell terminals (eg. VTxxx) via the WKU VMS Freeware file server – see question Section 13.1 for details.

3.2 What online information and websites are available?

On your OpenVMS system, the HELP command can provide a wealth of information, not only on DCL commands but on system services (HELP System_Services) and Run-Time Library routines (HELP RTL_Routines). The introduction displayed when you type the HELP command with no additional keywords provides further pointers.

OpenVMS Marketing runs a WWW server at http://www.openvms.compaq.com/. Here, you will find product information, strategy documents, product roadmaps, the contents of the latest OpenVMS Freeware CD-ROM and more.

Table 3–1   OpenVMS Websites

URL	Sponsor
http://www.openvms.compaq.com/	
http://www.openvms.compaq.com/wizard/	
http://www.openvms.compaq.com/wizard/wizard.zip	
	HP OpenVMS Marketing
http://www.montagar.com/	
	Encompass DFWCUG
http://www.levitte.org/~ava/	
	Arne Vajhøj
http://www.saiga.com/	
	Saiga Systems
http://www.tachysoft.com/	
	Wayne Sewel
http://www.progis.de/openvms.htm	
	Sponsored by proGIS Software
http://www.jcameron.com/vms/	
	Jeff Cameron
http://seqaxp.bio.caltech.edu/www/soft_doc.html	
	David Mathog's (useful) information about OpenVMS.

Cracking

http://www.vistech.net/users/beave/hack-vms-faq

"The Beave"
Includes system cracking information that can be of interest to
OpenVMS System Managers, and to OpenVMS Network and
Security Managers.

Undocumented Features

http://www.decus.de:8080/www/vms/qaa/undoc.htmlx

DECUS Deutchland

http://www.levitte.org/~ava/vms_tip.htmlx

Arne Vajhøj

http://www.openvms.compaq.com/freeware/

Table 3–1 (Cont.)   OpenVMS Websites

URL	Sponsor

The OpenVMS Freeware contains various examples of undocumented features and interfaces

Bibliographies

http://www.levitte.org/~ava/vms_book.htmlx

Introductory

http://www.levitte.org/~ava/vms_faq.htmlx

http://seqaxp.bio.caltech.edu/www/vms_sheet.html

http://seqaxp.bio.caltech.edu/www/vms_beginners_faq.html

Programming

http://www.pdv-systeme.de/users/martinv/VMS_Programming_FAQ.html

An OpenVMS Programming FAQ

Networking

http://www.tmesis.com/internet/

Tutorial information and tips for connecting OpenVMS systems to the Internet

http://www.openvms.compaq.com/wizard/decnet/

Documentation and Specifications for DECnet Phase IV

HP OpenVMS Documentation

http://www.openvms.compaq.com:8000/
http://www.openvms.compaq.com/doc/
http://www.openvms.compaq.com/commercial/

Various introductory guides as well as more advanced manuals are available in the OpenVMS and layered product documentation set.

http://www.compaq.com/info/spd/

Software Product Descriptions (SPDs) for most every OpenVMS-related product HP sells.

System Performance

See Section 14.2.

Patch (ECO) Kits

For the HP Services FTP server hosting Various contract-access and non-contract access ECO (patch) kits, see section Section 5.16.

Table 3–1 (Cont.)   OpenVMS Websites

URL	Sponsor
Catalogs and Pricing	
http://www.compaq.com/products/quickspecs/productbulletin.html	
	HP Product QuickSpecs and product information
http://www.compaq.com/products/quickspecs/soc_archives/SOC_Archives.html	
	The HP Systems and Options Catalog (SOC) archive
http://www.businesslink.compaq.com/	
	Pointers to country-specific product information, pricing, and related. The services formerly provided by BusinessLink are being replaced by these and other country-specific mechanisms, please see the URL for details.
Publications	
http://www.openvms.compaq.com/openvmstimes/	
	The OpenVMS Times Newsletter
http://www.dfwcug.org/	
	The DFWCUG Quadwords Newsletter
http://www.research.compaq.com/wrl/DECarchives/DTJ/	
	Back issues of the (discontinued) Digital Technical Journal (DTJ)
http://www.compaq.com/inFORM/	
	The HP (Compaq) inFORM Magazine
Hardware and Software Archives	
http://vax.sevensages.org/index.html	
	The VAXarchive, including hardware and software information
http://www3.sympatico.ca/n.rieck/docs/alpha_diary.html	
	A VAX to Alpha upgrade diary
http://www.montagar.com/~patj/dec/hcps.htm	
	Scanned versions of old DIGITAL manuals from DFWCUG
http://www.digital.com/lists/master-index.html	
http://www.compaq.com/support/techpubs/qrg/index.html	
	A wide variety of HP VAX, Alpha, platform and other product documentation. Some introductory, some technical.

3.3   OpenVMS Product Information Telephone Numbers?

Information on HP hardware, software, products and services is available through telephone numbers listed in Table 3–2:

Table 3–2   Telephone Numbers

Telephone	Description
1-800-AT-COMPAQ	HP (Compaq, including DIGITAL and Tandem) products and services
1-800-STORWORK	The HP StorageWorks team

3.4   How do I extract the contents of a HELP topic to a text file?

To extract all the text of a HELP topic (and its subtopics) to a text file for perusal with a text editor, printing out, etc., use the following command:

```
$ HELP/OUT=filename.txt help-topic [help-subtopic]
```

If the help text you want is not in the standard help library (for example, it's help for a utility such as MAIL that has its own help library), add /LIBRARY=libname after the HELP verb. To see the names of help library files, do a directory of SYS$HELP:*.HLB.

3.5   Does OpenVMS Marketing have an e-mail address?

Yes - if you can't get the answers to marketing questions elsewhere, if you have comments or complaints about OpenVMS, send mail to openvms-info@compaq.com. This address is *not* a support channel, and is solely intended to provide informal method to communicate directly with members of OpenVMS Marketing.

3.6   Where can I learn about OpenVMS executive internals?

The OpenVMS Internals and Data Structure manual (IDSM) explains how the OpenVMS executive works. The book covers the operating system kernel: process management; memory management; the I/O subsystem; and the mechanisms that transfer control to, from, and among these. It gives an overview of a particular area of the system, followed by descriptions of the data structures related to that area and details of the code that implements the area.

The first edition of the OpenVMS Alpha internals book describes Version 1.5. Although there have been several releases of OpenVMS Alpha since Version 1.5 (V6.1, V6.2, V7.0, V7.1, etc) and many details in the book are no longer accurate, it continues to provide a strong conceptual description of OpenVMS internals.

This book has been split into five pieces, each to be updated separately. The first such volume, published in early 1997, was "OpenVMS Alpha Internals and Data Structures: Scheduling and Process Control," which covers the Version 7.0 implementation of true multithreading and the changed scheduling model it implies.

The internals books are available through Digital Press, see Section 3.7

3.7 Where can new users find tutorial information about OpenVMS?

First, see if your local site has information on this topic. Each site can have site-specific features and configuration. Some sites will have site-specific new user's documentation, covering various site-specific things that are difficult or impossible for the general OpenVMS documentation to cover.

3.7.1 Tutorial Websites?

Various websites with OpenVMS information are available; Table 3–3 contains some suggested URLs.

Table 3–3   OpenVMS Tutorial Websites

URL	Sponsor
Introductory	
http://www.levitte.org/~ava/vms_faq.htmlx	
http://seqaxp.bio.caltech.edu/www/vms_sheet.html	
http://seqaxp.bio.caltech.edu/www/vms_beginners_faq.html	
	Various introductory materials
http://www.montagar.com/openvms_class/	

Table 3–3 (Cont.)   OpenVMS Tutorial Websites

URL	Sponsor
	Members of the Encompass DFWCUG maintain a website with many materials available, including an Overview of OpenVMS, an Introduction to DCL and the TPU Editor, Advanced DCL Command Procedures, OpenVMS Operations: Batch, Print, Tape, an Introduction to OpenVMS Management, to OpenVMS User Management, to OpenVMS Network Management, and to OpenVMS Cluster Management. These training materials have been presented at various DECUS symposia.

HP OpenVMS Documentation

http://www.openvms.compaq.com:8000/
http://www.openvms.compaq.com/doc/
http://www.openvms.compaq.com/commercial/

> Various introductory guides as well as more advanced manuals are available in the OpenVMS and layered product documentation set.

HP OpenVMS Training

http://www.compaq.com/training/home.html
http://www.openvms.compaq.com/wbt/index.html

> HP offers training information and Technical Resource Kits (TRKs) and other Training for OpenVMS. An OpenVMS certification (testing) program is also available.

http://www.jcameron.com/vms/

> An OpenVMS Quiz

http://www.CCSScorp.com/

> CCSS Interactive Learning has OpenVMS training materials

http://www.acersoft.com/

> AcerSoft Training information, and Shannon Knows Punditry

http://www.mindiq.com/

> MindIQ training information

3.7.2   Books and Tutorials?

Some of the OpenVMS books that are or have been available from the Digital Press imprint

* http://www.bh.com/

are listed in Table 3–4:

Table 3–4   DP Books

Title and Author	ISBN
Introduction to OpenVMS, 5th Edition Lesley Ogilvie Rice	1 55558 194 3
Introduction to OpenVMS David W Bynon	1 878956 61 2
OpenVMS Alpha Internals: Scheduling and Process Control	1 55558 156 0
OpenVMS AXP Internals and Data Structures: Version 1.5	1 55558 120 X
OpenVMS System Management Guide Richard Berry	1 55558 143 9
The OpenVMS User's Guide, Second Edition Patrick Holmay	1 55558 203 6
Using DECwindows Motif for OpenVMS Margie Sherlock	1 55558 114 5
VAX/VMS Internals and Data Structures: Version 5.2	1 55558 059 9
Writing Real Programs in DCL, Second Edition Hoffman and Anagnostopoulos	1 55558 191 9

For various featured OpenVMS books, also please see:

- http://www.openvms.compaq.com/openvms/books.html

For a bibliography of various OpenVMS books, please see:

- http://www.levitte.org/~ava/vms_book.htmlx

3.8   What OpenVMS mailing lists are available?

Various OpenVMS mailing lists are available, with some of the available lists detailed in Table 3–5.

Table 3–5   OpenVMS Mailing Lists

Subscription	Interest Area
OpenVMS Freeware archive announcement list	FSupdate@goatley.com
	FSupdate-request@goatley.com[1]
Two-way echo of vmsnet.internals	VMSnet-Internals@goatley.com
	VMSnet-Internals-request@goatley.com[1]
OpenVMS Alpha Internals discussions	Alpha-IDS@goatley.com
	Alpha-IDS-request@goatley.com[1]
BLISS discussions	BLISSters@goatley.com
	BLISSters-request@goatley.com[1]
Process Software MultiNet mailing list (news gateway)	Info-MultiNet@process.com
	Info-MultiNet-request@process.com[1]
Process Software TCPware mailing list (news gateway)	Info-TCPware@process.com
	Info-TCPware-request@process.com[1]
Process Software PMDF mailing list (news gateway)	Info-PMDF@process.com
	Info-PMDF-request@process.com[1]

[1]This is the subscription address. Usually, you will want to send a mail message with no subject line, and a SUBSCRIBE or HELP command in the body of the mail message.

Table 3–5 (Cont.)   OpenVMS Mailing Lists

Subscription	Interest Area
The SRI CHARON-VAX VAX emulator package	CHARON-VAX-Users@process.com
	CHARON-VAX-Users-request@process.com[1]
Info-Zip's Zip & UnZip discussion list	Info-Zip@wku.edu Info-Zip-Request@wku.edu[1]
RADIUS-VMS, a RADIUS server for OpenVMS discussion forum	radius-vms@dls.net radius-vms-request@dls.net[1]
Internet Service Providers (ISPs) running OpenVMS	vms-isps@dls.net vms-isps-request@dls.net[1]
Users of Mark Daniel's WASD web server for OpenVMS VAX and Alpha exists. Information about this list server and details on how to subscribe to the list are available at the referenced website.	http://wasd.vsm.com.au/

[1]This is the subscription address. Usually, you will want to send a mail message with no subject line, and a SUBSCRIBE or HELP command in the body of the mail message.

3.9   What is this Ask The Wizard website I've heard about?

The HP OpenVMS Ask The Wizard (ATW) website is an informal area containing questions and answers on a wide variety of topics.

- http://www.openvms.compaq.com/wizard/

- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

Before posting a question to the Ask The Wizard area, please read and please heed the posting rules.

3.10 Access to the OpenVMS Netscape Navigator documentation?

The documentation URLs embedded into the browser itself may not operate correctly in all cases, and (for reasons not worthy of repeating here) redirects may not be available.

You can manually access the documentation via:

- http://www.openvms.compaq.com:88/netscape/help/

4 Time and Timekeeping

4.1 UTC vs GMT vs vs UT1/UT1/UT2 TDF? What are these acronyms?

The results of an international compromise—though some would say an international attempt to increase confusion—UTC is refered to as "Coordinated Universal Time" (though not as CUT) in English and as "Temps Universel Coordinné" (though not as TUC) in French. (No particular information exists to explain why UTC was chosen over the equally nonsensical TCU, according to Ulysses T. Clockmeister, one of the diplomats that helped establish the international compromise.)

Universal Time UT0 is solar time, UT1 is solar time corrected for a wobble in the Earth's orbit, and UT2 is UT1 corrected for seasonal rotational variations in rotation due to the Earth's solar orbit.

GMT—Greenwich Mean Time—is UT1. GMT is the time at the classic site of the since-disbanded Royal Greenwich Observatory; at the most widely-known tourist attraction of Greenwich, England.

UTC is based on an average across multiple atomic clocks, and is kept within 0.9 seconds of GMT, through the insertion (or removal) of seconds. In other words, UTC matches GMT plus or minus up to 0.9 seconds, but UTC is not GMT.

TDF is the Timezone Differential Factor, the interval of time between the local time and UTC. Areas that celebrate daylight savings time (DST) will see periodic changes to the TDF value, when the switch-over between daylight savings time and standard time occurs. The switch-over itself is entirely left to local governmental folks, and can and has varied by political entity and politics, and the switch-over has varied over the years even at the same location.

If your local OpenVMS system time is off by one hour (or whatever the local DST change) for some or all applications, you probably need to reset your local TDF. (For related details, please see sections Section 4.4 and Section 10.23.0.0.0.1.) x>(Daylight Savings Time)

Further discussions of history and politics, the Royal Observers' outbuildings, and the compromise that left the English with the Time Standard (the Prime Meridian) and the French with the standards for

Distance and Weight (the Metric System) are left to other sources. Some of these other sources include the following URLs:

- ftp://elsie.nci.nih.gov/pub/

- http://physics.nist.gov/GenInt/Time/time.html

- http://nist.time.gov/

4.2 A brief history of OpenVMS Timekeeping, please?

Why does OpenVMS regards November 17, 1858 as the beginning of time...

The modified Julian date adopted by the Smithsonian Astrophysical Observatory (SAO) for satellite tracking is Julian Day 2400000.5, which turns out to be midnight on November 17, 1858.

SAO started tracking satellites with an 8K (nonvirtual) 36-bit IBM 704 in 1957 when Sputnik went into orbit. The Julian day was 2435839 on January 1, 1957. This is 11225377 octal, which was too big to fit into an 18-bit field. With only 8K of memory, the 14 bits left over by keeping the Julian date in its own 36-bit word would have been wasted. SAO also needed the fraction of the current day (for which 18 bits gave enough accuracy), so it was decided to keep the number of days in the left 18 bits and the fraction of a day in the right 18 bits of one word.

Eighteen bits allows the truncated Julian Day (the SAO day) to grow as large as 262143, which from November 17, 1858, allowed for 7 centuries. Possibly, the date could only grow as large as 131071 (using 17 bits), but this still covers 3 centuries and leaves the possibility of representing negative time. The 1858 date preceded the oldest star catalogue in use at SAO, which also avoided having to use negative time in any of the satellite tracking calculations.

The original Julian Day (JD) is used by astronomers and expressed in days since noon January 1, 4713 B.C. This measure of time was introduced by Joseph Scaliger in the 16th century. It is named in honor of his father, Julius Caesar Scaliger (note that this Julian Day is different from the Julian calendar that is named for the Roman Emperor Julius Caesar!).

Why 4713 BC? Scaliger traced three time cycles and found that they were all in the first year of their cyle in 4713 B.C. The three cycles are 15, 19, and 28 years long. By multiplying these three numbers (15 * 19 * 28 = 7980), he was able to represent any date from 4713 B.C. through 3267 A.D.

4–2

The starting year was before any historical event known to him. In fact, the Jewish calendar marks the start of the world as 3761 B.C. Today his numbering scheme is still used by astronomers to avoid the difficulties of converting the months of different calendars in use during different eras.

The following web sites:

- http://www.openvms.compaq.com/openvms/products/year-2000/leap.html

- http://www.eecis.udel.edu/~ntp/

- http://www.nist.gov/

- http://www.boulder.nist.gov/timefreq/

- http://www.tondering.dk/claus/calendar.html

- http://es.rice.edu/ES/humsoc/Galileo/Things/gregorian_calendar.html

are all good time-related resources, some general and some specific to OpenVMS.

4.2.1 Details of the OpenVMS system time-keeping?

4.2.1.1 VAX hardware time-keeping details...

4.2.1.1.1 TOY clock

This is battery backed up hardware timing circuitry used to keep the correct time of year during rebooting, power failures, and system shutdown. This clock only keeps track of months, days, and time. The time is kept relative to January 1st, at 00:00:00.00 of the year the clock was initiailized.

4.2.1.1.2 EXE$GQ_SYSTIME

This is the OpenVMS VAX system time cell. This cell contains the number of 100ns intervals since a known reference. This cell is incremented by 100000 every 10ms by an hardware interval timer.

4.2.1.1.3 EXE$GQ_TODCBASE

This cell contains the time and date the system time was last adjusted by EXE$SETTIME. It uses the same format as EXE$GQ_SYSTIME. On adjustment of the system time a copy of EXE$GQ_SYSTIME is stored in this cell in both memory and on disk. This cell is used to get the year for the system time.

4.2.1.1.4 EXE$GL_TODR

This cell contains the time and date the system time was last adjusted by EXE$SETTIME. It uses the same format as the time of year clock. On adjustment of the system time this cell gets saved back to both memory and disk. The contents of this cell are used to test the validity of the TOY clock.

The system parameters SETTIME and TIMEPROMPTWAIT determine how the system time will be set.

IF SETTIME = 0

THEN the contents of the TOY clock are compared to those of EXE$GL_TODR. IF the TOY clock is more than a day behind EXE$GL_TODR

THEN the TOY clock is presumed invalid.

- IF the TOY clock is within a day of EXE$GL_TODR
  THEN the system time is calculated as follows:

- EXE$GQ_SYSTIME = EXE$GQ_TODCBASE + ((TOY_CLOCK - EXE$GL_TODR) * 100000)

- IF SETTIME = 1 or the TOY clock is invalid
  THEN the value of TIMEPROMPTWAIT determines how to reset the time of year. IF TIMEPROMPTWAIT > 0
  THEN the user is prompted for the time and date, for a length of time equal to TIMEPROMPTWAIT microfortnights.

  - IF TIMEPROMPTWAIT = 0
    THEN the time of year is the value of EXE$GL_TODR + 10ms.

  - IF TIMEPROMPTWAIT < 0
    to proceed until they do so.

  - THEN the user is prompted for the time and date and unable

When booting a CD-ROM containing an OpenVMS VAX system, the system will typically be deliberately configured prompt the user to input the time – this is necessary in order to boot with the correct time.

If either TIMEPROMPTWAIT or SETTIME are set to zero, OpenVMS VAX will use the TOY clock to get the time of year, and the year will be fetched from the CD-ROM. The value of the year on the CD-ROM media (saved within the SYS.EXE image) will most likely be that of when the CD-ROM was made, and cannot be changed. Unless the current year happens to be the same year as that on the CD-ROM, most likely the year will be off. (Further, with the calculation of Leap Year also being

dependent on the current year, there is a possibility that the date could be off too.)

4.2.1.2   Alpha hardware time-keeping details...

4.2.1.2.1   Battery-Backed Watch (BB_WATCH) Chip

This is battery backed up hardware timing circuitry used to keep the correct time of year during rebooting, power failures, and system shutdown. This clock keeps track of date and time in 24 hour binary format.

4.2.1.2.2   EXE$GQ_SYSTIME

This is the OpenVMS Alpha system time cell. This cell contains the number of 100ns intervals since November 17, 1858 00:00:00.00. This cell is incremented by 100000 every 10ms by an hardware interval timer.

4.2.1.2.3   EXE$GQ_SAVED_HWCLOCK

This cell is used by OpenVMS Alpha to keep track of the last time and date that EXE$GQ_SYSTIME was adjusted. It keeps the same time format as EXE$GQ_SYSTIME. The value in this cell gets updated in memory and on disk, every time EXE$GQ_SYSTIME gets adjusted.

- The system parameters SETTIME and TIMEPROMPTWAIT determine how the system time will be set.

- If SETTIME = 0 then EXE$INIT_HWCLOCK reads the hardware clock to set the system time.

  - IF TIMEPROMPTWAIT > 0
    THEN the value of TIMEPROMPTWAIT determines how long the user is prompted to enter the time and date. If time expires and no time has been entered the system acts as if TIMEPROMPTWAIT = 0.

  - IF TIMEPROMPTWAIT = 0
    THEN the system time is calculated from the contents of EXE$GQ_SAVED_HWCLOCK + 1.

  - IF TIMEPROMPTWAIT < 0
    THEN the user is prompted for the time and date and unable to continue until the information is entered.

Unlike the VAX, the Alpha hardware clock tracks the full date and time, not just the time of year. This means it is possible to boot from the CD-ROM media without

entering the time at the CD-ROM bootstrap. (This provided that the time and date have been initialized, of course.)

IA-64 (Itanium) hardware time-keeping details to be added...

4.2.1.3 Why does VAX need a SET TIME at least once a year?

Because the VAX Time Of Year (TOY) has a resolution of 497 days, the VAX system time is stored using both the TOY and the OpenVMS VAX system image SYS.EXE. Because of the use of the combination of the TOY and SYS.EXE, you need to issue a SET TIME command (with no parameters) at least once between January 1st and about April 11th of each year, and whenever you change system images (due to booting another OpenVMS VAX system, booting the standalone BACKUP image, an ECO that replaces SYS.EXE, etc).

The SET TIME command is automatically issued during various standard OpenVMS procedures such as SHUTDOWN, and it can also obviously be issued directly by a suitably privileged user. Issuing the SET TIME command resets the value stored in the TOY, and (if necessary) also updates the portion of the time (the current year) saved in the SYS.EXE system image.

This VAX TOY limit is the reason why OpenVMS VAX installation kits and standalone BACKUP explicitly prompt for the time during bootstrap, and why the time value can "get weird" if the system crashes outside the 497 day window (if no SET TIME was issued to update the saved values), and why the time value can "get weird" if a different SYS$SYSTEM:SYS.EXE is used (alternate system disk, standalone BACKUP, etc).

4.2.2 How does OpenVMS VAX maintain system time?

VAX systems maintain an interval clock, and a hardware clock.

The VAX hardware clock is called the TOY ("Time Of Year") clock. The register associated with the clock is called the TODR ("Time Of Day Register").

The TOY clock—as used—stores time relative to January first of the current year, starting at at 00:00:00.00. It is a 100 Hz, 32-bit counter, incremented every 10ms, and thus has a capacity of circa 497 days.

OpenVMS (on the VAX platform) stores system date information—
and in particular, the current year—in the system image,
SYS$SYSTEM:SYS.EXE.

The TOY is used, in conjunction with the base date that is stored and
retrieved from the system image, to initialize the interval clock value that
is stored in EXE$GQ_SYSTIME.

Once the interval clock is loaded, the system does not typically reference
the TOY again, unless a SET TIME (with no parameters) is issued. The
interval clock value is updated by a periodic IPL22 or IPL24 (depending
on the specific implementation) interrupt. (When these interrupts are
blocked as a result of the activity of higher-IPL code—such as extensive
driver interrupt activity or a hardware error or a correctable (soft)
memory error—the clock will "loose" time, and the time value reported to
the user with appear to have slowed down.)

On most (all?) VAX systems, the battery that is associated with the
TOY clock can be disconnected and replaced if (when) it fails—TOY clock
failures are quite commonly caused by a failed nickel-cadmium (NiCd) or
lithium battery, or by a failed Dallas chip.

4.3   Keeping the OpenVMS system time synchronized?

To help keep more accurate system time or to keep your system clocks
synchronized, TCP/IP Services NTP, DECnet-Plus DECdtss, DCE DTSS,
and other techniques are commonly used. If you do not have IP access
to a time-base, then you could use dial-up access to NIST or other
authoritative site.

There exists code around that processes the digital (ie: binary) format
time that is available via a modem call into the NIST clock (the
Automated Computer Telephone Service (ACTS)), and code that grabs the
time off a GPS receiver digital link, or a receiver (effectively a radio and
a codec) that processes the time signals from radio station WWV, WWVH,
WWVB, or similar. (Processing these time protocols often involves little
more than reading from an EIA232 (RS232) serial line from the receiver,
something that is possible from most any language as well as directly
from DCL.)

One example of acquring a time-base involves the IRIG time format
(IRIG-A, -B, -G), a binary signal containing the current time in hours,
minutes, seconds and days since the start of the current year. IRIG can
also contain the time of day as the number of seconds since midnight.

HP Custom Systems and third-party vendors offer various IRIG-based reader/generator modules for OpenVMS systems.

Differing time servers (DECnet-Plus DTSS, DCE DTSS, NTP, etc) do not coexist particularly well, particularly if you try to use all these together on the same node. Please pick and use just one. (If needed, you can sometimes configure one package to acquire its timebase from another protocol, but one and only one time server package should have direct control over the management of and drifting of the local OpenVMS system time. In the specific case of DECnet-Plus DTSS, older product versions and versions V7.3 and later provide a provider module, a module which permits DTSS to acquire its time from NTP. For details on this, please see the comments in the module DTSS$NTP_PROVIDER.C.)

Useful URLs:

- http://www.boulder.nist.gov/timefreq/service/nts.htm
- http://www.boulder.nist.gov/timefreq/service/acts.htm
- http://www.boulder.nist.gov/timefreq/
- http://www.time.gov/

4.3.1 Why does my OpenVMS system time drift?

x>(TIME3)

Memory errors, hardware problems, or most anything operating at or above IPL 22 or IPL 24 (clock IPL is system family dependent; code executing at or above the clock IPL will block the processing of clock interrupts), can cause the loss of system time. Clock drift can also be caused by normal (thermal) clock variations and even by the expected level of clock drift.

When clock interrupts are blocked as a result of the activity of high-IPL code—such as extensive driver interrupt activity or a hardware error or a correctable (soft) memory error—the clock will "loose" time, and the time value reported to the user with appear to have slowed down. Correctable memory errors can be a common cause of system time loss, in other words.

Clock drift can also be (deliberately) caused by the activity of the DTSS or NTP packages.

Also see Section 14.14, Section 14.29, and Section 4.3.2.

4–8

4.3.2　How can I drift the OpenVMS system time?

With DECdts and TCP/IP Services NTP, the system time value is "drifted"
(rather than changed), to avoid the obvious problems that would arise
with "negative time changes". The same basic clock drifting technique is
used by most (all?) time servers operating on OpenVMS, typically using
the support for this provided directly within OpenVMS.

An example of the technique used (on OpenVMS VAX) to drift the system
time is the SETCLOCK tool on the OpenVMS Freeware.

For information on the use of the EXE$GL_TIMEADJUST and EXE$GL_
TICKLENGTH cells on OpenVMS Alpha, see *OpenVMS AXP Internal and
Data Structures*, located on page 348.

For those areas which switch between daylight savings time (DST) and
standard time, the time value is *not* drifted. The time is adjusted by the
entire interval. This procedure is inherent in the definition of the switch
between DST and standard time.

4.3.3　How can I configure TCP/IP Services NTP as a time provider?

An NTP time provider provides its idea of the current time to NTP clients
via the NTP protocol. Most systems are NTP clients, but...

NTP has a heirarchy of layers, called strata. The further away from the
actual NTP time source (Internet time servers are at stratum 1), the
lower the strata (and the larger the number assigned the statum).

NTP explicity configured at stratum one provides time to NTP operating
at lower strata, and the provided time is acquired based on the local
system time or via some locally-accessable external time source.

NTP at other (lower) strata both receive time from higher strata and can
provide time to lower strata, and automatically adjust the local stratum.
The highest stratum is one, and the lowest available stratum is fifteen.

The TCP/IP Services NTP package can operate at any stratum, and can
be configured as a peer, as a client, or as a broadcast server. NTP can also
provide time to a DECnet-Plus DTSS network, see Section 4.3.

With TCP/IP Services V5.0 and later, the only supported reference clock
is the LCL (local system clock). If your system has an excellent clock or
if the system time is being controlled by some other time service (such
as DTSS or GPS), you can configure NTP to use the system clock as its
reference source. This will mimic the master-clock functionality, and will

configre NTP as a stratum 1 time server. To do this, enter the following commands in TCPIP$NTP.CONF:

```
server 127.127.1.0 prefer
fudge 127.127.1.0 stratum 0
```

For local-master functionality, the commands are very similiar. Use:

```
server 127.127.1.0
fudge 127.127.1.0 stratum 8
```

The difference between these two is the stratum, and the omission of the prefer keyword. Specifying a higher stratum allows the node to act as a backup NTP server, or potentially as the sole time server on an isolated network. The server will become active only when all other normal synchronization sources are unavailable. The use of "prefer" causes NTP to always use the specified clock as the time synchronization source.

With the TCP/IP Services versions prior to V5.0, the NTP management is rather more primitive. To configure the local OpenVMS system from an NTP client to an NTP server (on TCP/IP Services versions prior to V5.0), add the following line to the sys$specific:[ucx$ntp]ucx$ntp.conf file:

```
master-clock 1
```

Also, for TCP/IP Services prior to V5.0, see the NTP template file:

```
SYS$SPECIFIC:[UCX$NTP]UCX$NTP.TEMPLATE
```

Note that NTP does *not* provide for a Daylight Savings Time (DST) switch-over, that switch must arise from the timezone rules on the local system and/or from the SYS$EXAMPLES:DAYLIGHT_SAVINGS procedure. (Further, there is a known bug in SYS$EXAMPLES:DAYLIGHT_SAVINGS.COM in V7.3, please obtain the available ECO kit.)

For current TCP/IP Services and related OpenVMS documentation, please see:

- http://www.openvms.compaq.com:8000/
- http://www.openvms.compaq.com/doc/
- http://www.openvms.compaq.com/commercial/

4.4   Managing Timezones, Timekeeping, UTC, and Daylight Savings?

You will want to use the command procedure:

- SYS$MANAGER:UTC$TIME_SETUP.COM

to configure the OpenVMS Timezone Differential Factor (TDF) on
OpenVMS V6.0 and later. Select the BOTH option. This configures
the OpenVMS TDF settings, though it may or may not configure the TDF
and the timezone rules needed or used by other software packages. Please
do NOT directly invoke the following command procedures:

- SYS$MANAGER:UTC$CONFIGURE_TDF.COM ! do not directly use

- SYS$MANAGER:UTC$TIMEZONE_SETUP.COM ! do not directly use

TCP/IP Services V5.0 and later use the OpenVMS TDF, UTC, and
timezone support. Earlier versions use a TDF mechanism and timezone
database that is internal to the TCP/IP Services package. Also on the
earlier versions, the TDF must be manually configured within TCP/IP
Services, in addition to the OpenVMS configuration of the TDF.

DECnet-Plus in V7.3 and later uses the OpenVMS TDF, UTC, and
timezone support, and displays its timezone prompts using UTC$TIME_
SETUP.COM. Earlier versions use a TDF TDF mechanism, timezone
database, and automatic switch-over that is internal to the DECnet-Plus
package. Also on earlier versions, the TDF must be configured within the
DECnet-Plus DECdtss package, in addition to the OpenVMS configuration
of the TDF.

Application code using HP C (formerly Compaq C, formerly DEC C)
will use the OpenVMS UTC and TDF mechanisms when the C code is
compiled on OpenVMS V7.0 and later (and when the macro _VMS_V6_
SOURCE is NOT defined). HP C does NOT use the OpenVMS UTC and
TDF mechanisms when the C code is compiled on OpenVMS releases
prior to V7.0, or when the preprocessor declaration _VMS_V6_SOURCE is
declared.

DCE DTSS TDF details TDB.

In OpenVMS Alpha V6.1, V6.2, and V6.2-1Hx, the TDF value is written
to SYS$BASE_IMAGE.EXE. With OpenVMS Alpha V7.0 and later and
with OpenVMS VAX V6.0 and later, SYS$SYSTEM:SYS$TIMEZONE.DAT
contains the TDF. This means that OpenVMS Alpha systems will need
to have the TDF value reset manually—usually within SYSTARTUP_
VMS.COM—on reboots prior to V7.0.

During OpenVMS Bootstrap, the SYSINIT module reads SYS$TIMEZONE.DAT to acquire the TDF for use in the system global cell EXE$GQ_TDF. This is done to ensure that the system boots with a valid TDF (a value which may be zero). The UTC system services get the TDF from this cell. These services, as well as the HP C RTL, must have a valid TDF. (Prior to OpenVMS V7.3, if either DECnet-Plus or DECnet/VAX Extensions is configured and run, the image DTSS$SET_TIMEZONE.EXE is invoked and can override the TDF and timezone rule settings from SYSINIT or from UTC$TIME_SETUP.COM— this image runs even if DTSS is disabled. If the settings do not match (due to inconsistencies in timezone specification in UTC$TIME_SETUP.COM and NET$CONFIGURE.COM), DTSS will reset the values to match its definitions.)

Prior to OpenVMS V7.3, daylight savings time switchover is handled automatically only when DCE DTSS or DECnet-Plus DTSS is in use. In V7.3, OpenVMS can be configured to automatically switch over to daylight savings time, and also generates an event that interested applications can use to detect the switch-over between standard time and daylight time.

The manual switchover between daylight savings time and standard time is correctly accomplished via the SYS$EXAMPLES:DAYLIGHT_SAVINGS.COM command procedure procedure.

Note: NTP (alone) does *NOT* provide automatic switch-over.

Note: The DST switch-over does *NOT* drift the time value; the switch-over applies the entire difference as a unit.

If you switch the TDF or daylight savings time setting, you will also want to restart or reconfigure any time-sensitive applications (those not using the time differential factor (TDF) change event available in V7.3 and later). Examples of these applications include the need to restart the NFS client and (yes) NTP. (NTP will want to try to "drift" the time (see Section 4.3), and will find the daylight savings time switch-over to be far too large to "drift". Hence the NTP restart.) You can also use the (undocumented) TCP/IP Services (prior to V5.0) commands:

```
SET TIME/DIFF=[positive or negative TDF integer]
GENERATE TIME
```

to reset the value of the logical name UCX$TDF.

Prior to V7.3, the command:

```
MCR DTSS$SET_TIMEZONE MODIFY
```

4–12

can be used to modify the settings of the SYS$TIMEZONE_DAYLIGHT_
SAVING, SYS$TIMEZONE_DIFFERENTIAL, and SYS$TIMEZONE_
NAME system logical names based on the SYS$TIMEZONE_RULE.

The following are other TDF-related logical names used/available on
OpenVMS systems, with typical Daylight Savings and Standard Settings
for the US Eastern Time (ET) timezone.

```
$daylight_time:
$ DEFINE/SYSTEM/EXECUTIVE MAIL$TIMEZONE EDT
$ DEFINE/SYSTEM/EXECUTIVE NOTES$TIMEZONE "-0400 EDT"
$ DEFINE/SYSTEM/EXECUTIVE LISP$DAYLIGHT_SAVING_TIME_P true  ! Not 'EDT'
$ DEFINE/SYSTEM/EXECUTIVE LISP$TIME_ZONE 05   ! Constant
$
$standard_time:
$ DEFINE/SYSTEM/EXECUTIVE MAIL$TIMEZONE EST
$ DEFINE/SYSTEM/EXECUTIVE NOTES$TIMEZONE "-0500 EST"
$ DEFINE/SYSTEM/EXECUTIVE LISP$DAYLIGHT_SAVING_TIME_P false ! Not 'EST'
$ DEFINE/SYSTEM/EXECUTIVE LISP$TIME_ZONE 05   ! Constant
$
$ DEFINE/SYSTEM/EXECUTIVE UCX$NFS_TIME_DIFFERENTIAL -
    'f$integer(f$element(0," ",f$logical("notes$timezone"))/-100)'
```

For information on ZIC and related tools used to manage the OpenVMS
Timezone database, please see the DEC C Run-time Library Utilities
Reference Manual—though the title would imply otherwise, this
particular manual is part of the OpenVMS documentation set, and not
part of the HP C (formerly Compaq C, formerly DEC C) documentation
set.

4.4.1   How to troubleshoot TDF problems on OpenVMS?

This is an OpenVMS Alpha system prior to V7.0 and the startup is not
invoking the procedure:

```
SYS$MANAGER:UTC$TIME_SETUP.COM
```

This is an OpenVMS system prior to V6.0, where there is no OpenVMS
TDF nor UTC available.

The version of the application does not use the OpenVMS TDF. This
includes TCP/IP Services prior to V5.0, applications using HP C built on
or targeting OpenVMS prior to V7.0, and systems using the DECnet-Plus
DTSS mechanisms prior to the release associated with OpenVMS V7.3.
(DCE TDF TBD.)

If you should find either of the following two timezone-related database files located in SYS$SPECIFIC:[SYSEXE]:

- SYS$SPECIFIC:[SYSEXE]SYS$TIMEZONE.DAT

- SYS$SPECIFIC:[SYSEXE]SYS$TIMEZONE_SRC.DAT

These two files are in an erroneous location and must be recreated in the correct directory:

```
SYS$COMMON:[SYSEXE]
```

If the DCL command:

```
$ DIRECTORY SYS$SYSTEM:SYS$TIMEZONE*.DAT
```

shows these files in SYS$SPECIFIC:[SYSEXE], then delete them and use SYS$MANAGER:UTC$TIME_SETUP.COM to recreate them.

On OpenVMS versions prior to V7.3, if the file:

```
$ SYS$STARTUP:DTSS$UTC_STARTUP.COM
```

is present on your system, then you may need to invoke:

```
$ @SYS$UPDATE:DTSS$INSTALL_TIMEZONE_RULE.COM
```

to recreate the timezone files correctly. Invoke this command immediately after [re]executing SYS$MANAGER:UTC$TIME_SETUP.COM.)

If SYS$UPDATE:DTSS$INSTALL_TIMEZONE_RULE.COM is not present on your system, then you may need to execute the following commands:

```
$ DELETE SYS$STARTUP:DTSS$UTC_STARTUP.COM
$ DEASSIGN/SYSTEM/EXEC SYS$TIMEZONE_RULE.
```

If your system time is being reported as being off by one hour (or whatever the local DST change), please see sections Section 4.1, Section 4.4 and Section 10.23.0.0.0.1. x>(Daylight Savings Time)

4.4.2 Customizing your TDF (Timezone) Setting?

Individual, local, and regional differences on the use (or the lack of use) of Daylight Savings Time (DST) are quite common.

If you need to add (or remove) daylight savings time for your area or otherwise alter the rules for your local area, you will probably end up creating a variation to an existing timezone rule.

4–14

The necessary zone line to add for WhereEverLand will probably look something like this:

```
# Zone  NAME             GMTOFF   RULES/SAVE       FORMAT  [UNTIL]
Zone    WhereEver        2:00     -                WhereEver
```

The OpenVMS source file for the timezone rules here:

```
SYS$COMMON:[SYS$ZONEINFO.SYSTEM.SOURCES]
```

You'll then want to ZIC this to create your own timezone definiton.

ZIC is documented in the OpenVMS Documentation Set, in the HP C Run-Time Library Reference Manual. (Despite the name of the manual, it is part of the OpenVMS documentation set and not the C manuals.)

Once you have created the new rule, use SYS$MANAGER:UTC$TIME_ SETUP.COM to select the new timezone—with V7.3 and later, this tool will notice the new timezone and will offer it, on earlier releases, you may/will have to hack the tool somewhat. (Don't even think of trying to define the TZ or other time-related logical names directly yourself.)

For various timezone rules, see the tar.gz files (these are gzipped tar archives) available at:

• ftp://elsie.nci.nih.gov/pub/

4.5   Why does the SET TIME command fail? Help managing DTSS?

If you try to set the system time with the SET TIME command, and see one of the following messages:

```
%SET-E-NOTSET, error modifying time
-SYSTEM-F-IVSSRQ, invalid system service request
```

```
%SET-E-NOTSET, error modifying time
-SYSTEM-E-TIMENOTSET, time service enabled; enter a time service command to update th
```

This occurs if the time on the local system is controlled by a time service software, for example the distributed time service software (DTSS) provided as part of the DECnet-Plus installation. The DTSS software communicates with one or more time servers to obtain the current time. It entirely controls the local system time (for DECnet-Plus, there is a process named DTSS$CLERK for this); therefore, the usage of the SET TIME command (and the underlying $SETTIM system service) is disabled.

The first message is displayed on systems running DECnet-Plus V6.1 and earlier. On systems with newer DECnet-Plus software, the second (and more informative) message is given.

You shouldn't have to change the time manually - you should be doing this through the time server - but if you insist... you'll have to shutdown DTSS:

```
$ MCR NCL
NCL> DISABLE DTSS
NCL> DELETE DTSS
```

This will shutdown DTSS$CLERK. You may then change the system time as usual. To restart the DTSS software, type

```
$ @SYS$STARTUP:DTSS$STARTUP
```

You will need a number of privileges to ussue this command, and you must also be granted the NET$MANAGE identifer to shutdown and to restart DTSS.

If you wish to "permanently" disable DTSS on a system running DECnet-Plus, the above NCL sequence must be performed each time the system is bootstrapped. (On DECnet-Plus V7.3 and later, you can define a logical name to disable the DTSS startup.)

If DTSS is running and no time servers are configured, you can (and will) see the following messages at regular intervals:

```
%%%%%%%%%%  OPCOM   2-SEP-1999 19:41:20.29  %%%%%%%%%%
Message from user SYSTEM on UNHEDI
Event: Too Few Servers Detected from: Node LOCAL:.mynode DTSS,
        at: 1999-09-02-19:41:20.296-04:00Iinf
        Number Detected=0,
        Number Required=1
        eventUid   5FA70F4F-616E-11D3-A80E-08002BBEDB0F
        entityUid  DE9E97DE-6135-11D3-8004-AA000400BD1B
        streamUid  D6513A46-6135-11D3-8003-AA000400BD1B
```

You can either configure the appropriate number of time servers, or you can disable DTSS, or you can ignore it and (if OPCOM is set to write to the log via via the logical names in SYLOGICALS.COM/SYLOGICALS.TEMPLATE) clean out OPERATOR.LOG regularly.

You can also simply disable the display of these messages:

```
$ mcr ncl block event dispatcher outbound stream local_stream global filter -
    ((Node, DTSS), Too Few Servers Detected)
```

4–16

If you wish to disable the automatic TDF adjustment for daylight savings time (on OpenVMS versions prior to V7.3), you can use the command:

```
ncl> set dtss automatic TDF change = false
```

or alternatively, you can set the local timezone to one that does not include the automatic daylight savings time change-over.

OpenVMS V7.3 and later simplify time and timezone management.

5   System Management Information

5.1   What is an installed image?

The term "install" has two distinct meanings in OpenVMS. The
first relates to "installing a product", which is done with either
the SYS$UPDATE:VMSINSTAL.COM command procedure or the
POLYCENTER Software Installation (PCSI) utility (PRODUCT
command). The second meaning relates to the use of the INSTALL
utility, which is what concerns us here.

The INSTALL utility is used to identify to OpenVMS a specific copy of an
image, either executable or shareable, which is to be given some set of
enhanced properties. For example, when you issue the SET PASSWORD
command, the image SYS$SYSTEM:SETP0.EXE is run. That image
needs to have elevated privileges to perform its function.

The other important attribute is /SHARED. This means that shareable
parts of the image (typically read-only code and data) are loaded
into memory only once and are shared among all users on a system.
Executable images can be installed /SHARED as well as shareable library
images. (The term "shareable" has dual meanings here, too. See the
OpenVMS Programming Concepts Manual for further details.)

It's important to note that there is no such thing as "installing a shareable
image with privileges". The INSTALL utility will let you do it, but the
privileges you specify will be ignored. To have a callable routine run with
enhanced privileges that are not available to its caller, you must construct
your routines as "user-written system services" and install the shareable
image with the /PROTECT qualifier. See the *OpenVMS Programming
Concepts Manual* for more information on user-written system services.
Note also that in many cases the need to grant privileges to an image
can be replaced with the use of the "Protected Subsystems" feature that
grants a rights identifier to an image. See the *OpenVMS Guide to System
Security* for information on Protected Subsystems.

5.2 Are there any known viruses for OpenVMS?

Viruses are very common on PCs because the PC operating systems such as MS-DOS and MacOS do not implement any sort of scheme to protect the operating system or the file system against hostile action by programs. On these operating systems, any running program can subvert the operating system and take over the hardware, at which point it can do anything it wishes, including hiding copies of itself in other programs or in the file system.

This is unlikely on OpenVMS, Unix, and MVS for three reasons. First, the operating system runs in a privileged mode in memory that is protected against modification by normal user programs. Any old program cannot take over the hardware as it can on PC operating systems. Secondly, OpenVMS, Unix, and MVS have file systems that can be set up so that non-privileged programs cannot modify system programs and files on disk. Both of these protection schemes mean that traditional PC virus schemes don't work on these OSes. Third, typical applications and configurations tend to prevent the uncontrolled execution of untrusted code as part of email messages or web access.

It is possible for OpenVMS, etc., to be infected by viruses, but to do so, the program containing the virus must be run from a user account that has amplified privileges. As long as the system administrator is careful that only trusted applications are run from such accounts (and this is generally the case), there is no danger from viruses.

To protect against viruses and other attempts at system interference or misuse, follow the recommendations in the "OpenVMS Guide to System Security". You may also want to consider optional software products which can monitor your system for intrusion or infection attempts. Computer Associates (CA) offers various products in this area.

Rocksoft offers the Veracity data integrity tool (for info, send mail to demo@rocksoft.com).

5.3 How do I mount an ISO-9660 CD on OpenVMS?

ISO-9660 support was added in the following releases:

- OpenVMS VAX V6.0
- OpenVMS AXP V1.5

An add-on ISO-9960 kit was also available for OpenVMS VAX V5.5, V5.5-1, V5.5-2, and V5.5-2H4. This requires the installation of the F11CD kit from the InfoServer CD, from the Consolidated Distribution CD under the InfoServer area, Customer Support Center kit CSCPAT #1071012, or the F11CD ECO kit. (Upgrades to V6 and later are strongly recommended.)

By default, OpenVMS senses the specific type of media. If you are working with dual-format media—media that uses both the ODS-2 and ISO-9660 formats on the same CD-ROM—then MOUNT will first detect and then default to the ODS-2 format. If you wish to override this and explicitly mount the media using ISO-9660, use the command:

```
$ MOUNT/MEDIA_FORMAT=CDROM  device-name[:] [volume-label]
```

In most circumstances, you will not need nor will you want to include an explicit /MEDIA_FORMAT specification. For further information, please refer to the OpenVMS MOUNT Utility Manual. Particularly note the information on the MOUNT /MEDIA_FORMAT and /UNDEFINED_FAT qualifiers.

The MOUNT /UNDEFINED_FAT qualifier is of interest because ISO-9660 media can be mastered on a wide variety of operating system platforms, and these platforms do not necessarily support the semantics needed for files containing predefined record formats. The /UNDEFINED_FAT allows you to specify the default attributes for files accessed from volumes using the ISO-9660 format.

An example which works for most CD-ROMs is:

```
$ MOUNT/MEDIA_FORMAT=CDROM/UNDEFINED_FAT=STREAM:2048 DUA0: FREEWARE
```

This particular MOUNT command forces access to the CD-ROM media using the ISO-9660 volume structure, and the use of the MOUNT /UNDEFINED_FAT qualifier causes any file whose file attributes are "undefined" to be returned with "stream" attributes with a maximum record length 2048.

On OpenVMS, the ISO-9660 format is (internally) considered to be the ODS-3 file structure, while the High Sierra extensions to the standard are considered to be the ODS-4 file structure. The Rock Ridge extensions are not currently available on OpenVMS.

For details on ODS-1 and ODS-2 file specifications, see Kirby McCoy's *VMS File System Internals Manual* (published by Digital Press, but potentially out of print), and see:

• http://pdp-11.trailing-edge.com/www/ods1.txt

- http://www.openvms.compaq.com/freeware/freeware50/ods2/

5.4   How do I extract the contents of a PCSI kit?

A growing number of OpenVMS products are being provided in PCSI (POLYCENTER Software Installation) kits which are installed using the PRODUCT INSTALL command. These are alternatives to or replacement for VMSINSTAL kits which were BACKUP savesets. PCSI kits are not BACKUP savesets and are structured differently from VMSINSTAL kits.

If you want to extract product files from a PCSI kit, create a directory into which the kit should be expanded and use the following command:

```
$ PRODUCT COPY prodname /SOURCE=[where-the-kit-is] -
    /DEST=[destination-directory] /FORMAT=REFERENCE
```

A PCSI kit file has a file specification of the following form:

```
DEC-VAXVMS-FORTRAN-V0603-141-1.PCSI
```

In this example, "FORTRAN" is the "prodname". PCSI will expand the kit files into the directory you specify and subdirectories beneath such as [SYSEXE], [SYSLIB], etc., reflecting the eventual destination of files found there. Most of the actual product files (images, etc.) will be in the subdirectories. In the top-level directory will be a file with the file type PCSI$DESCRIPTION that specifies where various files should go. For more details, see the *POLYCENTER Software Installation Developer's Guide for OpenVMS*, which can be found in the OpenVMS documentation on the Consolidated Online Documentation CD-ROM.

5.5   I've forgotten the SYSTEM password - what can I do?

If you need to break into an OpenVMS system because you do not have access to any privileged passwords, such as the password to the SYSTEM username, you will need physical access to the system console, and you will need to perform a conversational reboot. Here are the steps:

1   Halt the system. Exactly how this is done depends on the specific system model: Depending on the model, this can involve pressing the HALT button, entering CTRL/P on the console, or pressing the BREAK key on the console.

2   At the console prompt, use a console command to boot into the SYSBOOT utility. (SYSBOOT allows conversational changes to system parameters.) The syntax for the conversational bootstrap

varies by system model—this typically involves specifying a flag of 1, for example:

On VAX:

```
B/1
B/R5:1
@GENBOO
```

On Alpha:

```
b -flags 0,1
```

If your system has a non-zero system root (such as root SYSE, shown here), you will have to use a console command such as the following:

On VAX:

```
B/E0000001
B/R5:E0000001
@<console media procedure name varies widely>
```

On Alpha:

```
b -flags e,1
```

If your system has a hardware password (various systems support a password that prevents unauthorized access to the console), you will need to know theis password and will need to enter it using the LOGIN command at the console. If you get an "Inv Cmd" error trying to perform a conversational bootstrap, and you do not have the hardware console password for the console LOGIN command, you are stuck—you will need to call for hardware service in order to reset the hardware console password. The syntax used for the console password mechanism varies.

3 Once at the SYSBOOT prompt, request that OpenVMS read the system startup commands directly from the system console, that the window system (if any) not be started, and that OpenVMS not record these particular parameter changes for subsequent system reboots:

```
SET/STARTUP OPA0:
SET WINDOW_SYSTEM 0
SET WRITESYSPARAMS 0
CONTINUE
```

4 At the $ prompt, the system will now be accepting startup commands directly from the console. Type the following two DCL commands:

```
$ SPAWN
$ @SYS$SYSTEM:STARTUP
```

The result of these two commands will be the normal system startup, but you will be left logged in on the console, running under a privileged username. Without the use of the SPAWN command, you would be logged out when the startup completes.

If necessary, you can skip the invocation of the system startup temporarily, and perform tasks such as egistering license PAKs or various other "single-user" maintenance operations.

5   Use the following commands to reset the SYSTEM password:

```
$ SET DEFAULT SYS$SYSTEM:  ! or wherever SYSUAF.DAT resides
$ RUN SYS$SYSTEM:AUTHORIZE
MODIFY SYSTEM /PASSWORD=newpassword
EXIT
```

These steps will change the SYSTEM password to the specified new newpassword password value.

Reboot the system normally—the SYSTEM password is now set to the specified value.

Some people will suggest a method using the UAFALTERNATE SYSGEN parameter. This approach is not always reliable and is not recommended, as there can easily be an alternate user authorization file configured on the system.

For further information on emergency startup and shutdown, as well as for the official OpenVMS documentation on how to change the SYSTEM password from the console in an emergency, please see the *OpenVMS System Manager's Manual* in the OpenVMS documentation set.

You can also use the conversational bootstrap technique shown above (the steps through Step 3) to alter various system parameters. At the SYSBOOT prompt, you can enter new parameters values:

```
SHOW MAXPROCESSCNT
SET . 64
CONTINUE
```

The "." is a shorthand notation used for the last parameter examined.

5–6

5.6 How do I change the node name of an OpenVMS System?

The first step is to get a BACKUP of the system disk before making any changes—use the system disk backup procedures as documented in the OpenVMS System Management Manual, making sure to use the procedures and commands appropriate for the system disk.

Changing the node name involves a number of steps—the node name tends to be imbedded in a number of different data files around the system.

- Update the SCSNODE in MODPARAMS.DAT, and then run AUTOGEN as far as the SETPARAMS phase. (Do not reboot yet.)

- Modify the DECnet node name. (NETCONFIG is the DECnet Phase IV tool, and NET$CONFIGURE is the DECnet-Plus tool.)

- Modify the IP node name. (The TCP/IP Services tool is UCX$CONFIG prior to V5.0, and is TCPIP$CONFIG in V5.0 and later releases.)

- Modify the host node name on the various queues in the queue database. (each queue has a host name, and it defaults to the SCS node name of the queue's host system. See the command INIT/QUEUE/ON=node for information.)

- Modify the node name saved in any application databases, or any local node-conditional operations present in the site-specific system startup, etc. (SEARCH for the node name, specifying all types of files.)

- Rename the SYS$NODE_oldnodename rightslist identifier to match the new name. (Do not change the binary value of this identifier.)

- Reset any license PAKs that are restricted to the old node name to the new node name.

- If the node name is part of a disk volume label, see Section 5.12.

- Reboot the node or—if in a VMScluster—reboot the whole VMScluster. (This tends to catch any errors immediately.)

There are likely a few other areas where the nodename will be stored.

If the system is configured in a VMScluster and you change *either* the SCSNODE or the SCSSYSTEMID—but *not* both values—then you will have to reboot the entire VMScluster. (The VMScluster remembers the mapping between these two values, and will assume that a configuration problem has occured if a mismatched pair appears, and will refuse to let a node with a mismatched pair join the VMScluster.)

To calculate the correct SCSSYSTEMID value, multiply the DECnet Phase IV area number by 1024, and add the DECnet Phase IV node number. For example, the SCSSYSTEMID value for a DECnet node with address 19.22 is 19478. ((19 * 1024) + 22 = 19478)

This may well have missed one or two configuration tools (or more!) that are needed at your site—the node name tends to get stored all over the place, in layered products, and in local software...

Also see Section 15.6.3 and Section 15.6.4.

5.7 Why doesn't OpenVMS see the new memory I just added?

When adding memory to an OpenVMS system, one should check for an existing definition of the PHYSICALPAGES (OpenVMS VAX) or PHYSICAL_MEMORY (OpenVMS Alpha) parameter in the SYS$SYSTEM:MODPARAMS.DAT parameter database, use a text editor to reset the value in the file to the new correct value as required, and then perform the following command:

```
$ @SYS$UPDATE:AUTOGEN GETDATA REBOOT FEEDBACK
```

This AUTOGEN command will reset various system parameters based on recent system usage (FEEDBACK), and it will reset the value for the PHYSICALPAGES parameter to the new value. It will also reboot the OpenVMS system.

PHYSICALPAGES and PHYSICAL_MEMORY can also be used to deliberately lower the amount of memory available for use by OpenVMS. This ability can be useful in a few specific circumstances, such as testing the behaviour of an application in a system environment with a particular (lower) amount of system memory available.

PHYSICALPAGES and PHYSICAL_MEMORY can be set to -1 (on OpenVMS Alpha) or (better and simpler) the entry can be removed from the MODPARAMS.DAT file, to indicate that all available memory should be used.

5.8 How do I change the text in a user's UIC identifier?

The text translations of the numeric User Identification Code (UIC) are based on identifiers present in the OpenVMS rightslist. Documentation on this area is included in the _Guide to OpenVMS System Security_ manual.

To control the identifiers shown for a user's UIC, you use AUTHORIZE. Each user has an associated group identifier, and an identifier specific to the user. And each user should have a unique UIC.

5–8

To alter the text of a user or group identifier, use commands such as:

```
$ RUN SYS$SYSTEM:AUTHORIZE
UAF> rename/ident oldgroupid newgroupid
UAF> rename/ident olduserid  newuserid
```

If you should find yourself missing an identifier for a particular user, you can add one for the user's UIC using a command such as:

```
UAF> add/ident/value=uic=[group,user] newuserid
```

The UIC user identifier text is assigned when the username is created, and is the text of the username. The UIC group group identifier is assigned when the first username is created in the UIC group, and the text is based on the account name specified for the first user created in the group. The value of this identifier is [groupnumber, 177777]. To add a missing group identifier, use an asterisk as follows:

```
UAF> add/ident/value=uic=[group,*] newgroupid
```

You may find cases where an identifier is missing from time to time, as there are cases where the creation of a UIC group name identifier might conflict with an existing username, or a user identifier might conflict with an existing group identifier. When these conflicts arise, the AUTHORIZE utility will not create the conflicting group and/or user identifier when the username is created.

You can can add and remove user-specified identifiers, but you should avoid changing the numeric values associated with any existing identifiers. You should also avoid reusing UICs or identifiers when you add new users, as any existing identifiers that might be present on objects in the system from the old user will grant the same access to the new user. Please see the security manual for details.

5.9   What are the OpenVMS version upgrade paths?

5.9.0.0.0.1   OpenVMS Alpha Upgrade (or Update) Paths

```
From V1.0, one can upgrade to V1.5.
From V1.5, or V1.5-1H1, one can upgrade to V6.1.
From V6.1, one can upgrade to V6.2.
From V6.1, or V6.2, one can upgrade to V7.0.
From V6.1, V6.2, V6.2-
1H(1,2,3), or V7.0, one can upgrade to V7.1.
From V6.2, one can update to V6.2-1H1, V6.2-1H2, or V6.2-1H3.
From V6.2, V6.2-1H(1,2,3), V7.1, V7.1-1H(1,2), or V7.2, to V7.2-
1
From V6.2, ... or V7.2, to V7.2-1H1, to 7.3
From V7.1, one can update to V7.1-1H(1,2), ... to V7.2-
```

```
1H1, to 7.3
          From V7.2-2, and V7.3, one can update to V7.3-1
```

Some typical OpenVMS Alpha upgrade (or update) paths are:

```
V1.0 -> V1.5 -> V6.1 -> (V6.2, V7.0, V7.1, V7.2, V7.3)
V1.5-1H1 -> V6.1 -> (V6.2, V7.0, V7.1, V7.2, V7.3)
V6.2 -> V6.2-1H3
V6.2 -> V7.2-1
V6.2 -> V7.3
V6.2-1H(1,2,3) -> V7.1
V6.2-1H(1,2,3) -> V7.2-1
V7.1 -> V7.1-2
V7.1 -> V7.2-1
V7.1-1H(1,2) -> V7.1-2
V7.1-1H(1,2) -> V7.2-1
V7.2 -> V7.2-1H1
V7.2 -> V7.3 -> V7.3-1
V7.2-2 -> V7.3
V7.2-2 -> V7.3-1
```

Note that OpenVMS Alpha V7.0 does not include support for hardware and/or configurations first supported in OpenVMS Alpha V6.2-1H1, V6.2-1H2, or V6.2-1H3; one must upgrade to OpenVMS VAX V7.1.

One cannot update directly to a V6.2-1Hx Limited Hardware Release (LHR) from any release prior to the baseline V6.2 release. The same prohibition holds for performing updates directly to V7.1-1Hx from any release prior to V7.1—this is not supported, and does not produce the expected results. The LHR kits can, however, be directly booted and can be directly installed, without regard to any operating system that might be present on the target disk.

OpenVMS Alpha updates for LHRs (through V7.1-1Hx) require the use of VMSINSTAL for the update. These LHR releases use PCSI for the installation, but not for the update. Non-LHR releases use PCSI for installs and upgrades.

OpenVMS Alpha V7.1-2 and later use PCSI for LHRs and for OpenVMS upgrades and for all OpenVMS ECO kit installations. VMSINSTAL OpenVMS ECO kits are not used on OpenVMS Alpha V7.1-2 and later. Prior to V7.1-2, VMSINSTAL-based ECO kits are used for OpenVMS.

5–10

5.9.0.0.0.2 OpenVMS VAX Release Upgrade Paths

```
From V5.0 through V5.4-3 inclusive, one can upgrade to V5.5.
From V5.5, V5.5-1, or V5.5-2HW, one can upgrade to V5.5-2.
From V5.5, V5.5-1, or V5.5-2, one can upgrade to V6.0.
From V5.5-2, V5.5-2H4, or V6.0, one can upgrade to V6.1.
From V6.0, or V6.1, one can upgrade to V6.2.
From V6.1, or V6.2, one can upgrade to V7.0.
From V6.1, V6.2, or V7.0, one can upgrade to V7.1.
From V6.1, one can upgrade to V7.3 (with VAXBACK ECO for V6.1).
```

Some typical OpenVMS VAX upgrade paths are:

```
V5.x -> V5.5 -> V6.0 -> V6.2 -> (V7.1, V7.2, V7.3)
V5.5-2HW -> V5.5-2
V5.5-2, or V5.5-2H4 -> V6.1 -> (V6.2, V7.0, or V7.1)
V6.1 -> V6.1 with VAXBACK ECO -> (V7.2, V7.3)
V6.2 -> V7.2
V6.2 -> V7.3
```

Note that OpenVMS VAX V6.0 does not include support for hardware and/or configurations first added in OpenVMS VAX V5.5-2H4, one must upgrade to OpenVMS VAX V6.1.

Note that OpenVMS VAX V5.5-2HW is a pre-release version of V5.5-2. Any system running it should be upgraded to V5.5-2, or later.

If you attempt a direct upgrade from OpenVMS VAX V6.1 to V7.2 or later without having first applied the VAXBACK ECO kit to your V6.1 system, you will receive an error message:

```
%BACKUP-E-INVRECTYP, invalid record type in save set
```

and the upgrade will fail. Acquire and apply the VAXBACK ECO kit for OpenVMS VAX V6.1. OpenVMS VAX V6.2 and later do not require an application of an ECO for an upgrade to V7.2 and later.

5.9.0.0.0.3 OpenVMS Cluster Rolling Upgrade Paths

Rolling Upgrades require multiple system disks. Rolling upgrades permit the OpenVMS Cluster to remain available while individual systems are being upgraded to a new OpenVMS release.

OpenVMS Cluster rolling upgrades for both OpenVMS VAX and OpenVMS Alpha may (will) have different, or additional upgrade requirements, and have requirements around which versions of OpenVMS can coexist in a OpenVMS Cluster than what is listed here.

See the OpenVMS Upgrade and Installation Manual for the particular release, and the OpenVMS Software Product Descriptions for OpenVMS and for OpenVMS Cluster software:

- http://www.compaq.com/info/spd/
  OpenVMS typically uses SPD 25.01.xx and/or SPD 41.87.xx.

for further details on the rolling upgrade, and for support information. The documentation for older releases of OpenVMS VAX includes various platform-specific manuals, manuals that include instructions that are specific to installing and upgrading on the platform.

5.9.0.0.0.4 OpenVMS Product Version and Support Information

For information on Prior Version Support (PVS) and Mature Product Support (including information on support end dates for OpenVMS and various layered products), please see:

- http://www.compaq.com/services/software/ss_mature.html

- http://www.compaq.com/services/software/ss_pvs_se_amap.html

- http://www.compaq.com/services/software/ss_mps_pvs_eur.html

For information on supported versions of layered products, and minimum required layered product versions, see:

- http://www.openvms.compaq.com/openvms/os/swroll/index.html

For information on the release history of OpenVMS, including information on the code names of various releases and the major features:

- http://www.openvms.compaq.com/openvms/os/openvms-release-history.html

Additional release history information, as well as a variety of other trivia, is available in the VAX 20th anniversary book:

- http://www.openvms.compaq.com/openvms/20th/vmsbook.pdf

5–12

5.9.0.0.0.5 OpenVMS Alpha Terminology

The following terms apply to OpenVMS Alpha upgrades and installations.

- Update
  Typically used for Limited Hardware Releases (LHR) releases.
  Performed via VMSINSTAL. Applies only to the OpenVMS release
  that the LHR is based on, or to an intermediate LHR. (eg: V7.1-1H2
  applies only to V7.1-1H1 and to V7.1, not to any other releases.)
  LHRs within a series are cumulative, containing all files and features
  of previous LHRs in the same series.

- Upgrade
  Performed via PCSI. Upgrades can typically be applied to a release-
  specific (and documented) range of prior OpenVMS releases.

- Install
  Performed via PCSI. With an installation, no existing version of the
  operating system is assumed present, nor are any files from any copy
  of the operating system might be present preserved, and the entire
  contents of the target disk are destroyed via a disk initialization.

- preserve
  Performed via PCSI. Otherwise similar to an installation, this option
  skips the disk reinitialization. User files on the target disk are
  preserved. Any existing operating system files on the target disk are
  clobbered.

- LHR
  Limited Hardware Release. LHRs are specific to and are targeted
  at new hardware configurations, and are not shipped to customers
  with support contracts. At least one LHR kit must be specifically
  acquired when purchasing new hardware, new hardware that is not
  (yet) supported by any mainline (non-LHR) release. LHRs have an
  "H" in the OpenVMS version string, indicating a "Hardware" release.

For minimum OpenVMS versions for various platforms, see Section 2.11.

5.10 Why do I have negative number in the pagefile reservable pages?

Seeing a negative number in the reservable pages portion of the SHOW
MEMORY/FULL command can be normal and expected, and is (even)
documented behaviour. A pagefile with a negative number of reservable
pages is overcommitted, which is generally goodness assuming that every
process with reserved pages does not try to occupy all of the reserved
pagefile space at the same time.

5–13

To understand how the pagefile reservation process works, think about how a traditional bank operates when accepting customer deposits and making loans. It's the same idea with the pagefile space. There is less money in the bank vault than the total deposits, because much of the money has been loaned out to other customers of the bank. And the behaviour parallels that of the pagefile down to the problems that a "run on the bank" can cause for banking customers. (Though there is no deposit insurance available for pagefile users.)

If all of the running applications try to use the reserved space, the system manager will need to enlarge the pagefile or add one or more additional pagefules.

To determine if the pagefile is excessively overcommitted, watch for "double overcommitment"—when the reservable space approaches the negatation of the available total space—and watch that the total amount of free space available in the pagefile remains adequate. If either of these situations arises, additional pagefile storage is required.

Additional pagefile information: Additional pagefiles can typically be created and connected on a running OpenVMS system. New processes and new applications will tend to use the new pagefile, and existing applications can be restarted to migrate out of the more congested pagefiles. Pagefiles are generally named PAGEFILE.SYS, and multiple pagefiles are generally configured on separate disk spindles to spread the paging I/O load across the available disk storage. When multiple pagefiles are present on recent OpenVMS versions, each pagefile file should be configured to be approximately the same total size as the other pagefiles.

For additional information on pagefile operations and related commands, see the system management and performance management manuals in the OpenVMS documentation set.

With OpenVMS V7.3 and later, the displays have been changed and these negative values are no longer visible.

5.11   Do I have to update layered products when updating OpenVMS?

The Software Public Rollout Reports for OpenVMS list the current and future availability of HP software products shipping on the OpenVMS Software Products Library kits (CDROM consolidations) for OpenVMS Alpha and/or OpenVMS VAX. Specifically, the required minimum versions for product support are listed.

Comprehensive Public Rollout Information, listing previous product versions as well as currently shipping versions, has been compiled into a separate set of reports. The product information is grouped to show Operating System support.

You may or may not be able to use older versions of local applications, third-party products, and various HP OpenVMS layered products with more recent versions of OpenVMS. User-mode code is expected to be upward compatible. Code executing in a privileged processor mode—typically either executive or kernel mode—may or may not be compatible with more recent OpenVMS versions.

These reports are updated regularly. Please see:

*   http://www.openvms.compaq.com/openvms/os/swroll/index.html

---

5.12   How do I change the volume label of a disk?

Dismount the disk, and mount it privately. If the disk is mounted by more than one node in an OpenVMS Cluster, dismount it from all other nodes. If this disk is an OpenVMS system disk, shut down all other nodes that are bootstrapped from this disk.

Issue the SET VOLUME/LABEL command, specifying the new label.

On OpenVMS V6.0 and later, issue the following PCSI command to reset the label information stored within the PCSI database to reflect the new disk volume label:

```
$ PRODUCT REGISTER VOLUME old-label device
```

Locate any references in the system startup (typically including the disk MOUNT commands) and any DISK$label references in application files, and change the references appropriately.

If this is a system disk (for the host or for a satellite), also check the DECnet MOP or LANCP boot database, as well as any references to the disk created by CLUSTER_CONFIG*.COM.

Remount the disk appropriately.

5.13 How can I set up a shared directory?

To set up a shared directory—where all files created in the directory are accessible to the members of specified group of users—you can use an access control list (ACL) and an identifier.

The following also shows how to set up a resource identifier, which further allows the disk resources to be charged to the specified identifier rather than each individual user. (If you don't want this, then omit the attributes option on the identifier creation and omit the entry added in the disk quota database.

Add an identifier using the AUTHORIZE utility:

```
ADD/IDENTIFER/ATTRIBUTES=RESOURCE groupidentifier
```

Grant the identifier to each user in the group using AUTHORIZE:

```
GRANT/IDENTIFIER groupidentifier username
```

If disk quotas are in use, add an entry via SYSMAN for each disk:

```
DISKQUOTA ADD groupidentifier/PERMQUOTA=pq/OVERDRAFT=od/DEVICE=ddcu:
```

Set the shared directory to have an ACL similar to the following using the SET SECURITY (V6.0 and later) or SET ACL (versions prior to V6.0) command:

```
(DEFAULT_PROTECTION,S:RWED,O:RWED,G,W)
(IDENTIFIER=groupidentifier,OPTIONS=DEFAULT,ACCESS=READ+WRITE+EXECUTE+DELETE)
(IDENTIFIER=groupidentifier,ACCESS=READ+WRITE+EXECUTE+DELETE)
(CREATOR,ACCESS=READ+WRITE+ACCESS+DELETE)
```

If there are files already resident in the directory, set their protections similarly. (The OPTIONS=DEFAULT, DEFAULT_PROTECTION, and CREATOR ACEs apply to directories.)

The default protection mask is used to establish the default file protection mask, this mask does not prevent the users holding the specified groupidentifier from accessing the file(s), as they can access the file via the explicit identifier granting access that is present in the ACL.

For further information, see the *OpenVMS Guide to System Security Manual*, specifically the sections on ACLs and identifiers, and resource identifiers.

5.14 Why do I get extra blank pages on my HP Printer?

For information on configuring telnet print symbiont, on device control libraries such as SYSDEVCTL.TLB, and for ways of dealing with the extra blank pages that can arise on various HP printers, please see the OpenVMS Ask The Wizard area, starting particularly with topic (1020):

- http://www.openvms.compaq.com/wizard/

- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

There are a variety of discussions of this and of related printing topics in the Ask The Wizard area.

Also see Section 5.33.

5.15 Configure ELSA GLoria Synergy or PowerStorm 300/350 graphics?

5.15.0.0.0.1 The ELSA GLoria Synergy

On OpenVMS Alpha V7.1-2, V7.2, and V7.2-1, acquire the appropriate GRAPHICS PCSI kit, and all prerequisite OpenVMS ECO kits:

- VMS712_GRAPHICS-V0300 or later

- VMS72_GRAPHICS-V0100 or later

- VMS712_GRAPHICS-V0300 or later

The ELSA GLoria Synergy is the PBXGK-BB.

On OpenVMS Alpha V7.2-1, the files necessary for this graphics controller are located in the distribution CD-ROM directory:

```
DISK$ALPHA0721:[ELSA.KIT]
```

Also check for any available (later) ECO kits.

An earlier kit (ALP4D20T01_071) (for V7.1, V7.1-1H1, and V7.1-1H2) was once available, but has been superceded and is not recommended. Use of V7.1-2 or later (and use of one the above GRAPHICS kits as required) is typically the best approach.

OpenVMS V7.2-1H1 and later should directly support the controller.

Additional information is available in topics (3419) and (5448) in the Ask The Wizard area:

- http://www.openvms.compaq.com/wizard/

5–17

- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

5.15.0.0.0.2   PowerStorm 300, PowerStorm 350

The PowerStorm 300 is the PBXGD-AC, while the PowerStorm 350 is the PBXGD-AE.

For support of the PowerStorm 300 and PowerStorm 350 graphics controllers, acquire and install the following available ECO kits:

For OpenVMS Alpha V7.1-2:

- DEC-AXPVMS-VMS712_P350-V0100–4 or later
- DEC-AXPVMS-VMS712_GRAPHICS-V0300–4 or later

For OpenVMS Alpha V7.2-1:

- DEC-AXPVMS-VMS721_P350-V0100–4 or later
- DEC-AXPVMS-VMS721_GRAPHICS-V0300–4 or later

5.15.0.0.0.3   PowerStorm 3D30, PowerStorm 4D20

PowerStorm 3D30, PowerStorm 4D20 information is available in Ask The Wizard topics including (2041):

- http://www.openvms.compaq.com/wizard/
- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

Support for the ELSA GLoria Synergy and the PowerStorm 300 and 350 controllers is expected to be integrated in the OpenVMS Alpha V7.3 and later releases.

5.16   How can I acquire OpenVMS patches, fixes, and ECOs?

You can acquire and download kits containing OpenVMS fixes (ECOs) for various releases, as well as related support information, via:

- http://search.service.digital.com/
- ftp://ftp.support.compaq.com/public/vms/
- http://ftp.digital.com.au/pub/ecoinfo/

- http://ftp/digital.com.au/cgi-bin/grep/

- http://askq.compaq.com/

The http://ftp.digital.com.au/pub/ecoinfo/ URL can be particularly useful, as it includes a search engine capable of returning the mandatory ECO kits for each release. Also see the information on required ECOs available from the support database, accessable via http://askq.compaq.com/. Specifically, search for articles with the words "incorporated" and "need to install" in the title.

You can subscribe to an email notification list at:

- http://www.support.compaq.com/patches/mailing-list.shtml

A quarterly distribution is also available on CD-ROM:

- QT-3CQAA-C8 OpenVMS Alpha

- QT-3CRAA-C8 OpenVMS VAX

For a list of OpenVMS ECO kits recently released, you can use:

- http://Eisner.DECUS.org/conferences/OpenVMS-patches_new_1.HTML

You can also sign up for ECO kit email notifications (Digest or individual notifications) directly from HP at:

- http://www1.service.digital.com/patches/mailing-list.html

Examples and ECO kit installation instructions are included in the cover letter. For available ECO kits, cover letters and other associated documentation, look in:

- ftp://ftp.service.digital.com/public/vms/axp/...

- ftp://ftp.service.digital.com/public/vms/vax/...

Do NOT attempt to install a VMSINSTAL-based OpenVMS ECO kit on OpenVMS Alpha V7.1-2 and later. While VMSINSTAL itself remains available, it is not used for OpenVMS Alpha ECO kits starting in OpenVMS Alpha V7.1-2. OpenVMS Alpha V7.1-2 and later use PCSI for OpenVMS ECO kits.

See Section 5.29 for information on ECO kit checksums.

5.17 How do I move the queue manager database?

To move the location of the queue database, the
SYS$QUEUE_MANAGER.QMAN$QUEUES and SYS$QUEUE_
MANAGER.QMAN$JOURNAL files, to a disk that is fast(er), has plenty
of free space, and that is not heavily used. If the queue database is on a
(busy) OpenVMS system disk, you can and probably should move it off the
system disk to another disk spindle.

To move the queue database:

1   Checkpoint the journal file. This reduces the file size to the in-
    memory database size. This will cause the noted delay.

    ```
    $ mcr JBC$COMMAND
    JBC$COMMAND> DIAG 0 7
    ```

2   Stop the queue manager

    ```
    $ STOP/QUEUE/MANAGER/CLUSTER
    ```

3   Backup the .QMAN$QUEUES and .QMAN$JOURNAL files from the
    present location for safety.

    ```
    $ backup SYS$COMMON:[SYSEXE]SYS$QUEUE_MANAGER.QMAN$*  DISK:[DIR]
    ```

4   Create a new directory for the queue database. Insure that this disk
    is accessible to all nodes that can run the queue manager. If the /ON
    list for the queue manager is "/ON=(*)", the disk must be available to
    all nodes in the cluster

    ```
    $ CREATE/DIR fast_disk:[qman]
    ```

5   Copy the .QMAN$QUEUES and .QMAN$JOURNAL files to the new
    directory

    ```
    $ copy SYS$COMMON:[SYSEXE]SYS$QUEUE_MANAGER.QMAN$*  fast_disk:[qman]
    ```

6   Delete the old queue database.

    ```
    $ DELETE SYS$COMMON:[SYSEXE]SYS$QUEUE_MANAGER.QMAN$*;*
    ```

7   Restart the queue manager pointing to the new location

    ```
    $ START/QUEUE/MANAGER fast_disk:[qman]
    ```

5.18 How do I delete an undeletable/unstoppable (RWAST) process?

"Undeleteable" jobs are usually "undeleteable" for a reason—this can track back to insufficient process quotas, to a kernel-mode error in OpenVMS or a third-party device driver, or to other odd problems.

These undeletable jobs typically become of interest because they are holding onto a particular resource (eg: tape drive, disk drive, communications widget) that you need to use... If the particular device supports firmware, ensure that the device firmware is current – TQK50 controllers are known for this when working with old firmware. (That, and the infamous "MUA4224" firmware bug.) If this device has a driver ECO kit available, acquire and apply it... If the particular relevent host component has an ECO, acquire and apply it.

Useful tools include SDA (to see what might be going on) and DECamds (which increase and thus potentially fix quota-related problems). (nb: Applications with quota leaks will obviously not stay fixed.)

If the stuck application is BACKUP, ensure you have the current BACKUP ECO and are directly following the V7.1 or (better) V7.2 or later process quota recommendations for operator BACKUP accounts. Quota details are in the *OpenVMS System Manager's Manual*.

If the firmware and ECO levels are current, the best approach is to take a system crashdump, and pass a copy of the dump file it along to whomever is maintaining the device driver for the particular device/widget/driver involved, with any details on how you got into this situation. (The reboot involved with taking the crashdump will obviously clear the problem.)

There was some kernel-mode code (typically for OpenVMS VAX) that can reset the device ownership field, but that is rather obviously only an interim solution—the real fix is avoiding the loss of the IRP, the process quota leak, or whatever else is "jamming up" this particular process...

5.19 How do I reset the error count(s)?

The system reboot is the only supported approach, but it is obviously undesirable in various situations—there is presently no supported mechanism to reset error counts once the error(s) have been logged.

As for an unsupported approach—and be aware of the potential for causing a system crash...

To reset the error count, one needs to determine the system address of the error count field. For a device, this is at an offset within the device's UCB structure. On VAX, the field is at an offset symbolically defined as UCB$W_ERRCNT. On Alpha, this field's offset is symbolically defined as UCB$L_ERRCNT. The former is a word in size; the latter is a longword. (Could it be that Alpha devices are more error prone? ;)

You now need to locate the system address of the UCB$%_ERRCNT field of the device you wish to reset. Enter SDA. In the following, you will see designations in {} separated by a /. The first item in braces is to be used on the VAX and the second item should be used on an Alpha. (ie. {VAX/Alpha})

```
$ ANALYZE/SYSTEM
SDA>  READ SYS${SYSTEM/LOADABLE_IMAGES}:SYSDEF.STB
SDA>  SHOW DEVICE <ddnc:>    ! device designation of device with error
SDA>  EVALUATE UCB+UCB${W/L}_ERRCNT
Hex = hhhhhhhh   Decimal = -ddddddddd         UCB+offset
```

Record the hexadecimal value 'hhhhhhhh' returned.

You can now exit from SDA and $ RUN SYS$SHARE:DELTA or do what I prefer to do, issue the following:

```
SDA> SPAWN RUN SYS$SHARE:DELTA
```

On both VAX and Alpha, the DELTA debugger will be invoked and will ident- ify itself. On Alpha, there will be an Alpha instruction decoded. For those unfamiliar with DELTA, it does not have a prompt and only one error message—Eh? (Well, for sake of argument, there might be another error produced on the console if you're not careful—aka. a system crash!)

If you are on a VAX, enter the command: [W

If you are on Alpha, enter the command: [L

These set the prevailing mode to word and longword respectively. Remember the UCB${W/L)_ERRCNT differences?

Now issue the command 1;M

DELTA will respond with 00000001

You are now poised to ZAP the error count field. To do so you need to enter the system address and view its contents. The format of the command to do this is of the form:

```
IPID:hhhhhhhh/
```

For an IPID, use the IPID of the SWAPPER process. It is always: 00010001

5–22

Thus, to ZAP the error count, you would enter:

```
00010001:hhhhhhhh/
```

When you enter the / SDA will return the content of the address hhhhhhhh. This should be the error count (in hexadecimal) of the device in question. If it is not, you did something wrong and I'd suggest you type a carriage return and then enter the command EXIT to get out of DELTA. Regroup and see where your session went awry.

If you entered your address correctly and the error count was returned as in the following example, you can proceed.

```
00010001:80D9C6C8/0001                          ! output on VAX    1 error

00010001:80D9C6C8/00000001                      ! output on Alpha  1 error
```

You can now ZAP the error count by entering a zero and typing a carriage return. For example:

```
00010001:80D9C6C8/0001 0 return         ! output on VAX    1 error

00010001:80D9C6C8/00000001 0 return     ! output on Alpha  1 error
```

Now type the command EXIT and a carriage return.

Alternatively, reboot the system.

5.20 How do I find out if the tape drive supports compression?

For various SCSI-based MK-class magnetic tape devices:

```
$ Devdepend2 = F$GETDVI("$n$MKcxxx:","DEVDEPEND2")
$ Comp_sup = %X00200000
$ Comp_ena = %X00400000
$ IF (Devdepend2.AND.Comp_sup).EQ.Comp_sup THEN -
    WRITE SYS$OUTPUT "Compression supported"
$ IF (Devdepend2.AND.Comp_ena).EQ.Comp_ena THEN -
    WRITE SYS$OUTPUT "Compression enabled"
```

5.21 Can I copy SYSUAF to another version? To VAX? To Alpha?

The format of the SYSUAF.DAT, RIGHTSLIST, and associated files are upward-compatible, and compatible across OpenVMS VAX and OpenVMS Alpha systems. (This compatibility is a a basic requirement of mixed-version OpenVMS Cluster configurations and OpenVMS upgrades—for specific support information, please see the OpenVMS Cluster rolling upgrade and mixed-version requirements.) That said, it's the contents of the SYSUAF and RIGHTSLIST files that will make this more interesting.

5–23

The same basic steps necessary for moving RIGHTSLIST and SYSUAF files to another node are rather similar to the steps involved in merging these files in an OpenVMS Cluster—see the appendix of the OpenVMS Cluster documentation for details of merging files. (You might not be merging the contents of two (or more) files, but you are effectively merging the contents of the files into the target system environment.)

Considerations:

- applications often hold SYSUAF or RIGHTSLIST open, meaning a system reboot is often the best way to activate new files.

- the meanings of the RESTRICTED and CAPTIVE flags settings on the UAF entries have changed over time.

- the new NET$PROXY.DAT file that is initially created based on the contents of the NETPROXY.DAT during the OpenVMS VAX V6.1 upgrade and during the OpenVMS Alpha V6.2 upgrade. This file is maintained in parallel with NETPROXY.DAT.

- the RIGHTSLIST identifier values and UIC values that end up scattered around the target system must be rationalized with the contents of the new RIGHTSLIST and SYSUAF files.

The lattermost case—resolving the identifier values—is often the most interesting and difficult part. If you find that an identifier value (or identifier name) from the source RIGHTSLIST collides with that of an identifier existing on the target system, you must first determine if the two identifiers perform the same function. In most cases, they will not. As such, you will have to find and chance all references to the identifier value(s) (or name(s)) to resolve the "collision".

If you encounter a collision, changing both of the identifier binary values (or names) involved in the collision to new and unique values can prevent security problems if you should miss a couple of identifiers embedded somewhere on the target system during the whole conversion process— rather than the wrong alphanumeric value for the identifier being displayed, you'll simply see the binary format for the identifier displayed, and no particular access will be granted. And any DCL commands or such that reference the old alphanumeric name will fail, rather than silently (and potentially erroneously) succeeding.

Similar requirements exist for UIC values, as these too tend to be scattered all over the system environment. Like the binary identifier values, you will find UIC values associated with disks, ACLs, queues, and various other structures.

5–24

For a list of the various files shared in an OpenVMS Cluster and that can be involved when relocating an environment from one node to another (or merging environments into an OpenVMS Cluster), please see the SYLOGICALS.TEMPLATE file included in OpenVMS V7.2 and later releases.

Procedures to extract the contents of a (potentially corrupt) queue database are provided on the OpenVMS Freeware (V5) and can be used to combine two queue databases together while shuffling files between OpenVMS Cluster hosts.

For related discussions of splitting a cluster into two or for removing a node from cluster (political divorce, etc), see topics (203), (767), (915) and others in the Ask The Wizard area:

- http://www.openvms.compaq.com/wizard/
- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

5.22 How do I delete (timeout) idle processes?

There is no such command integrated within OpenVMS, though there are (optional) timers available within certain terminal servers and similar devices, and there is an integrated time-of-day mechanism that provides control over when a user can access OpenVMS.

As for available tools, there are DECUS, freeware, and third-party tools known variously as "idle process killers" (IPK) or terminal timeout" programs. Examples include: Saiga Systems Hitman, Watchdog, MadGoat Watcher (via the MadGoat site or the OpenVMS Freeware), Kblock, the Networking Dynamics tool known as Assassin, and the Zap tool.

A related package (for DECwindows sessions) is xtermlock.

If the forgetful users are in an application menu environment, the menu can potentially be extended to provide this capability.

5.23 Do I need a PAK for the DECevent (HP Analyze) tool?

DECevent and HP (Compaq) Analyze are avalable to customers with support contracts. The PAK is required only for the advanced functions of DECevent, the basic bits-to-text translation of the error log does not require a license PAK. Ignore the prompt, in other words. (The PAK should be available to you if you have a hardware support contract or warrantee, and the PAK enables the use of the advanced error analysis and notification capabilities within DECevent.)

Please see the DECevent FAQ for additional details:

- http://www.support.compaq.com/svctools/decevent/DECevent_FAQ.html

The current version of the DECevent (HP (Compaq) Analyze) tool can be downloaded from:

- http://www.support.compaq.com/svctools/st-download.html

5.24 INITIALIZE ACCVIO and ANSI tape label support?

A change was made (back in 1988) to (as it was then known) VAX/VMS V5.1-1 that added support for the then-new ANSI X3.27-1987 magnetic tape label standard. Prior to the ANSI X3.27-1987 standard, the date field in the ANSI HDR1 record permits dates only as far as the end of Year 1999. With ANSI X3.27-1987, dates through Year 1999 and dates from Years 2000 to 2099 are permitted.

Versions of INIT.EXE and MTAACP.EXE from VAX/VMS releases prior to V5.1-1 will potentially have problems properly processing ANSI magnetic tapes when Y2K and later dates are involved—the DCL INITIALIZE command is known to encounter access violation (ACCVIO) errors.

The available solutions include upgrades, or setting the date back. Direct initialization of the tape with the new headers (via $qio) is also clearly possible, though the limitation within the old MTAACP.EXE magtape ACP image is not nearly so easy to bypass.

5.25 How do I recover from INSVIRMEM errors?

Prior to OpenVMS Alpha V7.0 and on all OpenVMS VAX releases, VIRTUALPAGECNT and PGFLQUOTA limit the amount of virtual address space that is available to each process.

Further limiting the amount of address space is the size of system space (S0 and S1 space). On OpenVMS Alpha versions prior to V7.0 and on all OpenVMS VAX releases, VIRTUALPAGECNT and MAXPROCESSCNT together determine the size of the page table data structures that occupy large tracts of system space. When no system virtual address space is available for the stuff that needs it—this includes the page tables, non-paged pool, and various other structures—then the values of VIRTUALPAGECNT and MAXPROCESSCNT cannot be increased.

In OpenVMS Alpha V7.0 and later, the page table data structures have been moved out of S0 and S1 space and into page table space. In OpenVMS Alpha V7.2 and later, certain large data structures found in non-paged pool (eg: lock management structures) have been moved into 64-bit space, thus freeing up room in non-paged pool and in S0 and S1 space (where non-paged pool resides) while also permitting much larger data structures.

5.26 How can I prevent a serial terminal line from initiating a login?

In SYSTARTUP_VMS.COM, issue the command:

```
$ SET TERMINAL/NOTYPEAHEAD/PERMANENT ddcu:
```

This will prevent any unsolicited terminal input on ddcu:, and this unsolicited input is what triggers JOB_CONTROL to start up LOGINOUT on the terminal. Once LOGINOUT starts up on the serial line, you can see interesting behaviour (eg: audits, process creations, etc) as LOGINOUT tries to "chat" with whatever device is hooked onto the remote end of the serial terminal line.

5.27 How does PCSI use the image BUILD_IDENT field?

The (undocumented) build ident field in an OpenVMS Alpha image header is 16 bytes long, and is used as a counted string of 0-15 characters (ie, a an .ASCIC string with count in byte 0) and was originally introduced to provide information for use by VMSINSTAL patch kits to determine whether an image should be replaced or not.

Starting with OpenVMS Alpha V7.1-2, OpenVMS Engineering uses the PCSI utility to package and install ECO kits for OpenVMS. PCSI uses the generation attribute (a 32-bit unsigned integer) specified for files in the product description file (PDF) of a PCSI kit as the basis for performing file conflict detection and resolution. When a product is installed, PCSI modifies the build ident field of Alpha image headers to store an encoded form of the generation number. It also looks at the build ident field of previously installed images to obtain the generation information for those files as input to the file conflict processing algorithm. (Only images have this field, obviously.)

PCSI interprets the build ident field of a previously installed image as follows:

- if the string length is 15, the 5th character is a hyphen, and the last ten characters are a ten digit number with leading zeros, then the last ten characters are treated as a valid generation number.

- for V7.1-2 through V7.2-1, inclusive, if the above test fails, the information is obtained from the PCSI product database.

- in releases after V7.2-1 and with current PCSI ECO kits, if the above test fails, an invalid generation number is treated as 0000000000 so that the ECO kit will simply replace the image rather than assuming the PCSI database is in error.

So, what will you see in the image identification displayed via the ANALYZE/IMAGE command?

For an image that has been built as part of an OpenVMS Engineering system build, you will generally see a build ID string in the format "X6TE-SSB-0000"—X6TE is the build number for the OpenVMS Alpha V7.2-1 release. This id format is used within the OpenVMS system build, and can generally only be seen associated with images that have not yet been processed via PCSI.

During the installation of V7.2-1, PCSI will modify the image header to have a build ident string of "X6TE-0050120000". During installation of an ECO kit containing this image with a generation number of 50130052, for example, PCSI would determine that 50130052 is greater than 50120000, and will replace the existing image on the target disk with the version of the image included in the ECO kit.

5.28 How can I tell what software (and version) is installed?

There is unfortunately no consistent nor single way to make this determination—this is one of the reasons that a move to PCSI installations is underway.

On OpenVMS Alpha, you can use VMSINSTAL.HISTORY and PRODUCT SHOW PRODUCT to determine what packages have been installed via the VMSINSTAL and PCSI tools, respectively.

To see which OpenVMS Alpha ECO kits have been applied, look in VMSINSTAL.HISTORY on OpenVMS Alpha prior to V7.1-2, and use PRODUCT SHOW PRODUCT/FULL on OpenVMS Alpha V7.1-2 and later.

On OpenVMS VAX, you can use PRODUCT SHOW PRODUCT and (for software that is installed via VMSINSTAL on V7.3 and later) in VMSINSTAL.HISTORY.

For products installed on OpenVMS VAX prior to V7.3 using VMSINSTAL, there is no reliable way to determine what products have been installed. If the product provides a RELEASE_NOTES file (as many do), you can look for the list of these files via DIRECTORY SYS$HELP:*.RELEASE_NOTES. Again, this approach is NOT reliable: some kits do not provide release notes, some system managers will install only the release notes, some system managers will delete release notes, and release notes for multiple versions can be present.

On most packages, you can generally use ANALYZE/IMAGE on one of the core images, looking at the image identification area. Some of the product-specific mechanisms available are:

- DQS DQS$VERSION logical name
- C CC/VERSION
- C++ CXX/VERSION
- TCP/IP TCPIP SHOW VERSION command

5.29   What file checksum tools are available for OpenVMS?

The undocumented DCL command CHECKSUM is the usual means, and provides a rather simple-minded checksum suitable to detect basic file corruptions. For information and an OpenVMS version of the MD5 checksum tool, see:

• http://www.support.compaq.com/svctools/md5-instructions.html

The OpenVMS Alpha ECO (patch) kit checksums available at the ECO website are determined using the following DCL command sequence:

```
$ CHECKSUM kitname.pcsi-dcx_axpexe
$ SHOW SYMBOL CHECKSUM$CHECKSUM
```

See Section 5.16 for information on acquiring OpenVMS ECO (patch) kits.

5.30   What (and where) is the OpenVMS Management Station?

For information and current kits for the OpenVMS Management Station (OMS), a PC-based tool that permits you to manage an OpenVMS system, please see:

• http://www.openvms.compaq.com/openvms/products/argus/

5.31   How to determine current disk fragmentation level?

The HP OpenVMS Disk File Optimizer (DFO) defragmentation package provides a fragmentation monitoring tool, and a DFO product authorization key (PAK) is not required for the fragmentation reporting tool:

```
$ DEFRAG SHOW/VOLUME ddcu:
```

The DFU tool available on the OpenVMS Freeware can generate a report on the disk fragmentation:

```
DFU> REPORT ddcu:
```

5.32 SYSBOOT-I-FILENOTLOC, Unable to locate SYS$CPU_ROUTINES?

A message at the OpenVMS Alpha bootstrap such as the following:

```
%SYSBOOT-I-FILENOTLOC, Unable to locate SYS$CPU_ROUTINES_1C02.EXE
%SYSBOOT-E-LDFAIL, failed to load execlet, status = 00000910
```

indicates that the particular OpenVMS Alpha release does not contain support for the target platform. In this case, OpenVMS does not recognize Alpha family 1C member 02 as a supported platform. A later version of OpenVMS might support the platform, or there might be no support on any release. Ensure that you have the most current firmware, and review the minimum version requirements for the platform.

The execlet load failure and other similar bootstrap status values can often be decoded using either of the following techniques:

```
$ exit %x910
%SYSTEM-W-NOSUCHFILE, no such file
$
$ x = f$message(%x910)
$ show symbol x
  X = "%SYSTEM-W-NOSUCHFILE, no such file"
$
```

Also see Section 14.20.

5.33 How can I customize the DCPS device control for a new printer?

To customize DCPS for an otherwise unsupported printer, you can try the following sequence:

- Extract the most closely-associated setup modules from the existing device control library, DCPS$DEVCTL.TLB. (For instance, you can probably extract and use the HP LaserJet 4000 series definitions for the HP LaserJet 4050 series. Each printer will vary, please consult the printer documentation for specifics and requirements.)

- rename each extracted setup module to a corresponding:

```
LPS$$UNRECOGNIZED_*
```

- Insert all of the above-renamed setup modules into a newly-created device control library specific to the new printer:

```
$ LIBRARY/TEXT/CREATE -
    SYS$COMMON:[SYSLIB]HP4050_DEVCTL.TLB
    LPS$$UNRECOGNIZED*
```

The above assumes the filename HP4050_DEVCTL.TLB, alter as required.

- Set up your DCPS startup procedures to include a search-list logical name such as:

```
$ DEFINE/SYSTEM/EXECUTIVE DCPS_HP4050_LIB  -
    SYS$LIBRARY:HP4050_DEVCTL.TLB, -
    SYS$LIBRARY:DCPS$DEVCTL.TLB
```

- Supply DCPS_HP4050_LIB as the library parameter in the queue startup for this printer, this is the P3 parameter to the command procedure SYS$STARTUP:DCPS$EXECUTION_QUEUE.COM.

- The HP4050_DEVCTL library may/will need to be recreated and modules re-edited and replaced with each DCPS upgrade, particularly if any modules are updated in the original library. You will also want to determine if the upgraded version of DCPS directly supports the particular printer.

- To customize the processing of file extensions within DCPS (to enable or disable graybar output, for instance), use the information available in:

```
SYS$LIBRARY:DCPS$FILE_EXTENSION_DATA_TYPE.DAT_DEFAULT
```

to create your own site-specific:

```
SYS$LIBRARY:DCPS$FILE_EXTENSION_DATA_TYPE.DAT
```

Also see Section 5.14.

5.34   Why do $GETDEV MOUNTCNT and SHOW DEVICE mount counts differ?

MOUNTCNT returns the local mount count, while SHOW DEVICE returns the cluster-wide mount count.

5.35   What software is needed for Postscript printers?

The NorthLake PrintKit (http://www.nls.com/) and DECprint Supervisor (DCPS; http://www.openvms.compaq.com/openvms/Print/print_sw_prods.html) are common choices for support of Postscript printers on OpenVMS.

You may also require the installation of an IP transport stack.

Also please see Section 15.2.2 and Section 15.2.3.

5–32

5.36 How do I remove a PCSI-installed patch (ECO) kit?

You cannot PRODUCT REMOVE a PCSI patch (ECO) kit.

In order to remove an ECO kit, PCSI would have to have copies of all the other version of the files from all other patches and products that previously were installed. This can clearly involve a large number of files and a large archive of old file versions and a substantial quantity of disk space. While removal is clearly theoretically possible, it is not currently implemented.

The following is the supported mechanism to remove a PCSI patch kit.

1 Execute a PRODUCT SHOW PRODUCE <product-name. /FULL command. The "maintenance" column (132 column width) shows the patches that have been installed. Keep a copy of this listing.

2 Acquire kits for all of the maintenance kits listed.

3 Re-install the prior FULL version of the product. This will remove all patch kits, setting to product back to "original" condition.

4 Re-install all the patches in the list from step 1, *EXCEPT* those which you have determined you do not want.

The above information also applies to PCSI PARTIAL kits.

5.37 SYSINIT-E, error mounting system device, status=0072832C

This message can arise during an OpenVMS system bootstrap...

```
%MOUNT-F-DIFVOLMNT, different volume already mounted on this device
```

For details and further information, use the DCL command:

```
$ HELP/MESSAGE /STATUS=%X72832C
```

5.38 Resolving License PAK Problems?

The PAK release date, the PAK termination date, and the PAK version are the usual culprits when a license product authorization key (PAK) check failure occurs.

The PAK termination date is the date when the license PAK will expire.

The PAK release date is the date of the most recent release date of the software package that will be permitted by the particular license PAK. (The release date check is analogous to a product version check.) The PAK version indicates the most recent product version that is permitted by the license.

Having multiple license PAKs registered (and active) can also cause problems if an expired PAK gets loaded. You will want to DISABLE license PAKs you do not wish to have loaded.

Other problems include a failure to register each PAK in all license databases throughout a multiple-system-disk cluster, with a consistent set of /INCLUDE lists specified across each of the duplicated PAKs.

Additionally, you could have an invalid LMF$LICENSE logical name defined. (If no LMF$LICENSE logical name is defined, the standard license database named SYS$SYSTEM:LMF$LICENSE.LDB will be used.)

You can display license failures by defining the following logical name:

```
$ DEFINE/SYS/EXEC LMF$DISPLAY_OPCOM_MESSAGE TRUE
```

Enable your terminal as a license operator (REPLY/ENABLE=LICENSE), define the LMF$DISPLAY_OPCOM_MESSAGE logical name, and then try the failing operation again. You should see one or more OPCOM messages displayed.

If you have the LMF$DISPLAY_OPCOM_MESSAGE logical name defined, you can (will?) see spurious license check failures—various products will check for multiple licenses, and a few products will check for PAKs that either have not yet been or will not be issued. Once you figure out which license has failed, you will want to deassign this logical name.

Note: that there is no license check failure does NOT indicate that the particular product or operation is permissible per the license.

To register a license PAK on a DECwindows system when DECwindows cannot start (because of an expired license or other licensing problem), follow the steps outlined in section MGMT5 up through step 4 (inclusive). Using the console—analogous to what is done in step 5 to access the OpenVMS AUTHORIZE utility—use the console to register the license PAKs.

5.39 Changing the OpenVMS Version Number?

Fool your friends, baffle your enemies, run the OpenVMS version of your choice!

On OpenVMS Alpha systems:

```
$ SET DEFAULT SYS$COMMON:[SYS$LDR]
$ RUN SYSVER
REPLACE V9.9
WRITE
$ EXIT
```

On OpenVMS VAX systems:

```
$ set default SYS$COMMON:[SYS$LDR]
$ copy SYS.EXE SYS.EXE_IN-CASE-I-FAIL
$ patch SYS.EXE
define sys$gq_version=800044b8
set mode ascii
!examine sys$gq_version
!examine sys$gq_version+4
deposit sys$gq_version   = "V9.9"
deposit sys$gq_version+4 = "    "
update
exit
$ Exit
```

Then reboot the system at your leisure.

5.40 How to prevent users from choosing obvious passwords?

To prevent users from selecting obvious passwords on OpenVMS, you will want to use the reserved password (password screening) mechanism. Effectively, you merge your list of reserved passwords into the existing reserved words database maintained by OpenVMS. (You can also then require all users to reset their passwords—via the pre-expired password mechanism—thus forcing users to select new passwords.) For details on the password screening mechanism, of the reserved password database (VMS$PASSWORD_DICTIONARY.DATA), and details of how to merge your list of prohibited passwords into the database, please see the associated chapter in the OpenVMS security manual. For details of the password expiration mechanism, see the AUTHORIZE command qualifier /PWDEXPIRED.

You can also implement a site-specific password filter with the information provided in the back of the OpenVMS Programming Concepts manual. The password filter permits you to establish particular and site-specific password requirements. For details, please see the system parameter LOAD_PWD_POLICY and the programming concepts manual, and see the examples in SYS$EXAMPLES:. (Examples and documentation on V7.3 and later reflect both platforms, the examples are found only on OpenVMS VAX kits on earlier releases. The capabilities have existed on both the VAX and Alpha platforms for some time now.)

To verify current passwords, you can also use a technique known to system crackers as the "dictionary attack"—the mechanism that makes this attack somewhat more difficult on OpenVMS is the hashing scheme used on OpenVMS, and the file protections used for the SYSUAF authorization database. Given a dictionary of words and the unprotected contents of the SYSUAF file, a search for obvious passwords can be performed. Interestingly, a "dictionary attack" also has the unfortunate side-effect of exposing the password to the user—while this is clearly the goal of a system cracker, authorized privileged and non-privileged system users should not know nor have access to the (cleartext) passwords of other users.

Accordingly, OpenVMS does not store the cleartest password. Further, OpenVMS uses a password hashing algorithm, not an encryption algorithm. This means that storage of a cleartext password is deliberated avoided, and the cleartext value is deliberately very difficult to obtain. The hash is based on a Purdy Polynomial, and the hash itself includes user-specific values in addition to the password, values that make the results of the password hash unique to each user.

Regardless of the use of a password hashing scheme, if a copy of your password file should become available to a system cracker, you will want to force all users to use new passwords immediately.

If you should require a user to verify a password, use the username, the user's salt value (this value is acquired via $getuai) and the user's specified cleartext password, and compare the resulting hashed value (using a call to $hash_password) against the saved hashed password value (this value also acquired via $getqui). For reasons of security, avoid saving a cleartext password value in any data files, and do not maintain the cleartext password in memory longer than required. (Use of $ACM on V7.3-1 and later is recommended.)

5–36

Kerberos authentication (client and server) is available on OpenVMS V7.3 and later. Integration of Kerberos support into various Compaq and into third-party products is expected.

External authentication is available in V7.3-1 and later, with support for user-written external authentication expected in V7.3-2 and later.

If you are simply looking for OpenVMS access and the SYSTEM and all other privileged passwords are forgotten or otherwise unavailable, please see section Section 5.5 and/or the OpenVMS documentation set.

Also please see the C2 guidelines in the OpenVMS security manual.

5.41 Please help me with the OpenVMS BACKUP utility?

5.41.1 Why isn't BACKUP/SINCE=BACKUP working?

If you are seeing more files backed up than previously, you are seeing the result of a change that was made to ensure BACKUP can perform an incrementation restoration of the files. In particular, if a directory file modification date changes, all files underneath it are included in the BACKUP, in order to permit incremental restoration should a directory file get renamed.

5.41.1.1 Why has OpenVMS gone through the agony of this change?

When a directory is renamed, the modified date is changed. When the restoration needs to restore the directory and its contents, and the restoration should not result in the restoration of the older directory name when a series of incremental BACKUPs are restored. Thus an incremental BACKUP operation needs to pick up all of the changes.

Consider performing an incremental restoration, to test the procedures. This testing was how OpenVMS Engineering found out about the problem that was latent with the old BACKUP selection scheme—the old incremental BACKUP scheme would have missed restoring any files under a renamed directory. Hence the change to the selection mechanisms mentioned in Section 5.41.1.

5.41.1.2 Can you get the old BACKUP behaviour back?

Yes, please see the /NOINCREMENTAL qualifier available on recent OpenVMS versions (and ECO kits). Use of this qualifier informs BACKUP that you are aware of the limitations of the old BACKUP behaviour around incremental disk restorations.

5.41.2 What can I do to improve BACKUP performance?

Use the documented commands in the manual for performing incremental BACKUPs. Use the documented incremental procedures. Don't try to use incremental commands in a non-incremental context.

Also consider understanding and then using /NOALIAS, which will likely be a bigger win than will anything to do with the incremental BACKUPs, particularly on system disks and any other disks with directory aliases.

See the OpenVMS V6.2 release notes for additional details.

5.41.3 Why is BACKUP not working as expected?

First, PLEASE READ THE BACKUP MANUAL.

Second, PLEASE GET THE CURRENT BACKUP ECO KIT.

BACKUP has a very complex interface, and there are numerous command examples and extensive user documentation available. For a simpler user interface for BACKUP, please see the documentation for the BACKUP$MANAGER tool.

As for recent BACKUP changes, oddities, bugs, etc:

- A change made in OpenVMS V6.2 WILL cause more files to be included into a file-based BACKUP saveset using /SINCE=BACKUP as all files underneath any directory with a sufficiently recent (selected) date will be included in the saveset. This change was deliberate and intentional, and was mandated by the need to provide a functional incremental restoration.

  Without the inclusion of these apparently-extra files, an incremental saveset can NOT be reliably restored.

- As part of the OpenVMS V6.2 change, the /SINCE command—without the specification of the =BACKUP keyword—selected more files than it should have. This is a bug. This bug has been remedied in the OpenVMS BACKUP source code and in some of (all of?) the BACKUP ECO kits.

When working with BACKUP, you will want to:

- Get the current BACKUP ECO kit and install it BEFORE you attempt to troubleshoot any problems.

- Learn about the /NOINCREMENTAL (new) and /NOALIAS (V6.2 and later) command qualifiers. The former qualifier returns to the pre-V6.2 behaviour of the /SINCE file selection mechanism, while the latter (specified with /IMAGE) reduces the replication of files on system disks and other disks with file alias and directory alias entries. Both of these can reduce the numbers of files that will be selected and thus included into the saveset.

When working with the BACKUP callable API:

- Build your applications with the most current BACKUP API available. Changes made to the V7.1-2 and V7.2 API were incompatible with the V7.1 and V7.2-1 and later APIs, and this incompatibility was repaired via a BACKUP ECO kit. Do NOT build your application with the versions of the BACKUP API that shipped with V7.1-2 and V7.2, as these are incompatible with the BACKUP API constants that were used on other versions.

5.41.4   How do I fix a corrupt BACKUP saveset?

BACKUP savesets can be corrupted by FTP file transfers and by tools such as zip (particularly when the zip tool has not been asked to save and restore OpenVMS file attributes or when it does not support OpenVMS file attributes), as well as via other means of corruptions.

If you have problems with the BACKUP savesets after unzipping them or after an FTP file transfer, you can try restoring the appropriate saveset attributes using the tool:

```
$ @RESET_BACKUP_SAVESET_ATTRIBUTES.COM
```

This tool is available on the OpenVMS Freeware (in the [000TOOLS] directory). The Freeware is available at various sites—see the Freeware location listings elsewhere in the FAQ—and other similar tools are also available from various sources.

In various cases, a SET FILE/ATTRIBUTES command can also be used. As the parameters of this command must be varied as the target BACKUP saveset attributes vary, this approach is not recommended.

Also see the "SITE VMS", /FDL, and various other file-attributes options available in various FTP tools. (Not all available FTP tools support any or all of these options.)

Browser downloads (via FTP) and incorrect (binary or ascii FTP transfer modes) are notorious for causing RMS file corruptions and particularly BACKUP saveset corruptions. You can sometimes help encourage the browser to select the correct FTP transfer type code (via RFC1738):

- ftp://host/urlname.ext;type=i ! request ftp image/binary transfer

- ftp://host/urlname.ext;type=a ! request ftp ascii/text transfer

You can also often configure the particular web browser to choose the appropriate transfer mode by default, based on the particular file extensions, using a customization menu available in most web browsers. You can select that the specific file extentions involved use the FTP binary transfer mode, which will reduce the number of corruptions seen.

5.41.5   How do I write a BACKUP saveset to a remote tape?

p>How to do this correctly was described at DECUS long ago. On the OpenVMS host with the tape drive, create the following SAVE-SET.FDL file:

```
RECORD
        FORMAT                  fixed
        SIZE                    8192
```

Then create BACKUP_SERVER.COM:

```
$ !
$ ! BACKUP_SERVER.COM - provide remote tape service for BACKUP.
$ !
$ set noon
$ set rms/network=16
$ allocate mka500 tapedev
$ mount/nounload/over:id/block=8192/assist tapedev
$ convert/fdl=SAVE-SET sys$net tapedev:save-set.
$ dismount/unload tapedev
$ stop/id=0
```

On the node where you want to do the backup, use the DCL command:

```
$ backup -
    srcfilespec -
    node"user pwd"::"task=backup_server"/block=8192/save
```

One area which does not function here is the volume switch; multi-reel or multi-cartridge savesets. Since the tape is being written through DECnet and RMS and the magtape ACP, BACKUP won't see the media switch and will split an XOR group across the reel boundary. BACKUP might well be willing to read such a multi-reel or multi-cartridge saveset (directly, not over the net) as the XOR blocks are effectively ignored until and unless

5–40

needed for error recovery operations. BACKUP likely will not be able to perform an XOR-based recovery across reel or cartridge boundaries.

Unfortunately BACKUP can't read tapes over the network because the RMS file attributes on a network task access look wrong; the attributes reported include variable length records.

6 Information on Mail

6.1 MAIL keeps saying I have new messages, but I don't. What do I do?

The count of new mail messages is kept separately from your mail folder in SYS$SYSTEM:VMSMAIL_PROFILE.DATA. It sometimes happens that this count differs from what's in your mail folder. If this happens, go into MAIL and repeat the READ/NEW command until you see no new mail messages. Then enter the command one more time. This will resynchronize the counters.

6.2 How do I send or read attachments in VMS MAIL?

Is there any way to send or read mail with files as attachments from VMS?

Not directly with the OpenVMS MAIL facility, but there are several other options:

- Install PINE, available commercially from Innosoft or free from Andy Harper. With PINE you can both send and receive MIME messages, if you have the appropriate viewers available.

    - http://www.process.com/

    - http://www.agh.cc.kcl.ac.uk/files/vms/pine-vms/

    - ftp://ftp2.kcl.ac.uk/pub/vms/pine-vms/

- If you are working from an X11 server use the OpenVMS version of Netscape Navigator or Mozilla. This option is ok for sending mail, but is not optimal for reading it, since Netscape will use POP and remove messages entirely the OpenVMS MAIL system, which is not generally what you want.

- MPACK/MUNPACK. To send a MIME mail, construct the message with attachments manually using MPACK. You cannot send the resulting file directly through MAIL because an extra blank header line will be inserted between your message and the OpenVMS MAIL headers, which will cause the message to appear as plain text in most mail programs. Some TCP/IP stacks provide a work around for this problem, and if that doesn't work, you should generally be able to

force the message directly into the SMTP port of your mail machine. Examples of both methods are in:

- http://seqaxp.bio.caltech.edu/pub/SOFTWARE/mmail.com

To read a MIME mail message, open it in MAIL, extract it to a file, then use MUNPACK to break out and decode the attachments. MPACK/MUNPACK tools are also available on OpenVMS Freeware V5.0.

- With OpenVMS V7.2 and later, use the MIME tool supplied with OpenVMS.

6.3 How can I block SMTP mail relay spam?

Use the anti-spam capabilities present in the TCP/IP Services V5.1 and later SMTP servers.

Use a firewall.

On earlier TCP/IP Services releases, some simple DCL can reportedly prevent relay SMTP spam. Use the UCX command SHOW SERVICE SMTP/FULL to find the directory containing the UCX$SMTP_RECV_ STARTUP.COM file, and insert the following DCL:

```
$        !
$        ! Block spam.
$        !
$        MY_ADDRESS_LONG[0,32]=F$INTEGER(F$TRNLNM("SYS$REM_NODE")-"::")
$        MY_ADDRESS=F$FAO("!UB.!UB.!UB.!UB",F$CVUI(0,8,MY_ADDRESS_LONG),-
          F$CVUI(8,8,MY_ADDRESS_LONG),F$CVUI(16,8,MY_ADDRESS_LONG),-
          F$CVUI(24,8,MY_ADDRESS_LONG))'"
$        MY_ADDRESS_REVERSE=F$FAO("!UB.!UB.!UB.!UB",-
          F$CVUI(24,8,MY_ADDRESS_LONG),F$CVUI(16,8,MY_ADDRESS_LONG),-
          F$CVUI(8,8,MY_ADDRESS_LONG),F$CVUI(0,8,MY_ADDRESS_LONG))'"
$        WRITE SYS$OUTPUT F$TIME()+" "+F$TRNLNM("SYS$REM_NODE")+MY_ADDRESS
$        UCX SHOW HOST 'MY_ADDRESS_REVERSE'.INPUTS.ORBS.ORG
$        IF $STATUS.EQ.1
$        THEN
$          WRITE SYS$OUTPUT "SPAM from relay rejected"
$          EXIT
$        ENDIF
$        UCX SHOW HOST 'MY_ADDRESS_REVERSE'.SPAMSOURCES.ORBS.ORG
$        IF $STATUS.EQ.1
$        THEN
$          WRITE SYS$OUTPUT "SPAM source relay rejected"
$          EXIT
$        ENDIF
$        !
$        ! Run receiver.
$        !
$        run sys$system:ucx$smtp_receiver.exe
$        goto exit
```

7 Information on Utilities

7.1 How do I play an audio CD on my workstation?

If you've installed the DECwindows examples, you'll find
DECW$CDPLAYER.C, .DAT, .EXE, .UIL, and .UID. Copy the .UID and
.DAT files to DECW$USER_DEFAULTS: (typically SYS$LOGIN:), define
the logical name DECW$CD_PLAYER to be the device name of your
CD-ROM drive (eg. DKA400:), give yourself PHY_IO and DIAGNOSE
privileges, and run the .EXE. (These privileges are required, as the access
to the CD-related extensions will require the use of the privilege-protected
IO$_DIAGNOSE I/O function code.) You can also install the image with
these privileges. See the source for additional details - note that the
comments regarding the need for SYSGEN CONNECT are no longer
applicable (at least as of VMS V5.5-2).

There's also SYS$EXAMPLES:CDROM_AUDIO.C and .EXE,
a non-Motif program, available on OpenVMS VAX, and
DECW$EXAMPLES:DECW$CDPLAYER.* on OpenVMS VAX and
OpenVMS Alpha.

The standard OpenVMS IDE SYS$DQDRIVER does not support the
necessary does not support the necessary IO$_DIAGNOSE function
required for access to audio CD media (on OpenVMS versions prior
to V7.3), but an updated SYS$DQDRIVER device driver (source code
and all) with this capability and with the source code of an updated
DECW$CDPLAYER CD audio player is available on the OpenVMS
Freeware website (www.openvms.compaq.com/freeware/dqdriver/),
and these updates are also included on OpenVMS Freeware V5.0,
and OpenVMS ECO kits containing newer versions of the driver are
available.

7.2 How do I access a Microsoft Windows floppy disk from OpenVMS?

The HP Advanced Server (formerly known as PATHWORKS) for
OpenVMS product includes an unsupported and undocumented utility
called PCDISK, and this tool can read and write various Microsoft MS-
DOS and Microsoft Windows FAT-format diskettes.

ProGIS in Germany sells a product called VMove which supports DOS files on many different device types. For more information, send mail to info@progis.de.

Engineering Software has a product called VAKSAT which will read, write, and erase files on MS-DOS FAT diskettes. Available for both VAX and Alpha. Contact ed@cityscape.co.uk for more information.

MadGoat PC Exchange (PCX) is a utility for copying files to and from MS-DOS and Microsoft Windows (FAT) format diskettes under OpenVMS, using an RX23 (3.5"), RX26 (3.5"), or RX33 (5.25") diskette drive. For 3.5" diskettes, high-density disks can be read or written; double-density disks are read-only. Only high-density disks are supported on the RX33.

- http://www.madgoat.com/

7.3 How do I play sound files on an AlphaStation? DECsound doesn't work.

The new AlphaStation systems use a different sound board (Microsoft Sound System) than the earlier DEC 3000 series systems, and DECsound, as supplied by DECwindows Motif, doesn't support this board. HP offers an optional product, Multimedia Services for OpenVMS:

- http://www.compaq.com/info/spd/
  OpenVMS typically uses SPD 25.01.xx and/or SPD 41.87.xx.

which provides a replacement DECsound for this card as well as many other features (an AVI and MPEG player, video capture support, etc.)

Ensoniq sound support is also available.

7.4 Why is DECmigrate not working with Fortran?

```
%OTS-F-INDATCOR internal data corrupted in Run-time Library
```

This error can arise with Fortran programs if you are running a recent version of OpenVMS Alpha, and are using DECmigrate to translate Fortran applications. The DECmigrate Run-Time Library attempts to support mixed translated-native I/O to the same unit by sharing the native Fortran RTL's internal data structures, and in OpenVMS 7.2 these structures changed and the translated RTL was not updated accordingly.

You can copy DEC$FORRTL.EXE from OpenVMS 7.1, copying it to some spare directory, and then defining the logical name DEC$FORRTL to point to it before running your translated application. Or rebuilding the

application to use the available native Fortran compiler. Or you can apply the current Fortran RTL kit, which has a fix for this.

See Section 13.11.

7.5   How do I read IBM EBCDIC tapes on OpenVMS?

IBM boxes can read ANSI-labeled ASCII magtapes. Fixed-length records and the DCL COPY command can be used to transfer text files around. Check the IBM documentation for details.

There exists various freeware around (TAPECOPY, ETAPE, TCOPY, MTEXCH) that can read and write EBCDIC tapes. Visit the DECUS website software archives search engine (via http://www.decus.org/), and search for "EBCDIC".

One source for ETAPE is:

•   http://www.ualr.edu/ftp/vms/ETAPE_SRC/

OpenVMS Freeware V5.0 includes this ETAPE tool, as well.

7.6   How can I patch an OpenVMS Alpha image?

Using the OpenVMS Freeware tool ZAP:

•   www.openvms.compaq.com/freeware/freeware50/rms_tools/

tell ZAP to read a block (bucket) of information based on the virtual block number (VBN), using X for hexadecimal. Dump yourself into the OpenVMS debugger with R2 pointing into the buffer, EXAMINE/INSTRUCTION as needed, alter the buffer as required, GO to get out of the debugger and back into ZAP, and use the ZAP W command to write the updated block.

8   DCL Details

8.1   How do I run a program with arguments?

The RUN command does not accept arguments. To pass arguments to a program, you must use what is called a "foreign command". For example:

```
$ unzip :== $disk:[dir]unzip.exe
$ unzip -?
```

The leading $ in the equivilence name for the symbol definition is what makes the DCL symbol a foreign command. If the device and directory are omitted, SYS$SYSTEM: is assumed.

Under OpenVMS V6.2 and later, DCL supports automatic foreign command definition via the logical name DCL$PATH:. An example of a definition of this logical name is:

```
$ DEFINE DCL$PATH SYS$DISK:[],ddcu:[mytooldir],SYS$SYSTEM:
```

DCL will first look for a command in the DCL command table, and if no match is found and if DCL$PATH is defined, it will then look for command procedures and executable images with filenames matching the command specified, in the directories specified via DCL$PATH. The first match found is invoked, and under OpenVMS, the DCL$PATH support will cause a command procedure to be activated in preference to an executable image.

For more information on foreign commands or on automatic foreign command support, see the *OpenVMS User's Manual*.

See also Section 10.2.

If you want to create a detached process that takes arguments from a command line, it must be run under the control of a command line interpreter (CLI) (typically DCL). This is done by placing the command line in a file, specifying SYS$SYSTEM:LOGINOUT.EXE as the image to run and the command file as the input. For example:

```
$ OPEN/WRITE CMD TEMP_INPUT.COM
$ WRITE CMD "$ MYCOMMAND arguments"
$ CLOSE CMD
$ RUN/DETACHED SYS$SYSTEM:LOGINOUT /INPUT=TEMP_INPUT.COM
```

Various OpenVMS library calls (such as lib$spawn(), cli$dcl_parse(), and the C library system() call) require access to a command line interpreter such as DCL to perform requested actions, and will not operate if a CLI is not available.

When a CLI is not available, these calls typically return the error status SS$_NOCLI. And as mentioned above, invoke the image LOGINOUT to cause a CLI (such as DCL) to be mapped into and made available in the context of the target process.

For examples of how TCP/IP Services sets up its foreign commands (which includes tools such as uuencode and uudecode), please see the DCL command procedure SYS$STARTUP:TCPIP$DEFINE_ COMMANDS.COM.

Also see Section 8.10.

8.2 How can I clear the screen in DCL?

The simplest way is the TYPE/PAGE NLA0: command.

You can set up a symbol to clear the screen in your LOGIN.COM:

```
$ CLS :== TYPE/PAGE NLA0:
```

8.3 Using REPLY/LOG from DCL? Disabling Console OPCOMs?

Your terminal must be enabled as an operator terminal before the REPLY/LOG command can be used, but a DCL procedure (batch command file, system startup, etc) does not have an associated terminal. To make this work, use the following sequence to enable the OPA0: console as the operator terminal, then the REPLY/LOG command will be accepted:

```
$ DEFINE/USER SYS$COMMAND _OPA0:
$ REPLY/LOG
$ DEFINE/USER SYS$COMMAND _OPA0:
$ REPLY/ENABLE
```

To disable the system console terminal (OPA0:) as an operator terminal, use the following command:

```
$ DEFINE/USER SYS$COMMAND _OPA0:
$ REPLY/DISABLE
```

8–2

Also see SYLOGICALS.COM (and SYLOGICALS.TEMPLATE) for information on configuring the behaviour of OPCOM, including the (default) use of the system console (OPA0:) as an operator terminial and the specific contents and behaviour of the system operator log file OPERATOR.LOG.

8.4   How do I generate a random number in DCL?

Here is a random number generator, just do a GOSUB RAND and the global symbol RANDOM will contain a randomly generated number. You can feed the generator a ceiling value (__CEIL) or a new seed (__SEED).

```
$! RAND - returns a positive random number ("RANDOM") between 0 and
$!        __CEIL - 1.
$! sharris-at-sdsdmvax.fb3.noaa.gov
$ RAND:
$
$ IF F$TYPE(__SEED) .EQS. ""
$ THEN
$     ! seed the random number generator, ...
$     __NOW = F$CVTIME()
$     __HOUR = 'F$EXTRACT(11,2,__NOW)'
$     __MINUTE = 'F$EXTRACT(14,2,__NOW)'
$     __SECOND = 'F$EXTRACT(17,2,__NOW)'
$     __TICK = 'F$EXTRACT(20,2,__NOW)'
$
$     __SEED == __TICK + (100 * __SECOND) + (6000 * __MINUTE) + -
         (360000 * __HOUR)
$     ! the generator tends to do better with a large, odd seed, ...
$     __SEED == (__SEED .OR. 1)
$     ! clean up, ...
$     DELETEX/SYMBOL __NOW
$     DELETEX/SYMBOL __HOUR
$     DELETEX/SYMBOL __MINUTE
$     DELETEX/SYMBOL __SECOND
$     DELETEX/SYMBOL __TICK
$ ENDIF
$
$ IF F$TYPE(__CEIL) .EQS. "" THEN __CEIL = %X3FFFFFFF
$
$ __SEED == __SEED * 69069 + 1
$
$ RANDOM == (__SEED.AND.%X3FFFFFFF)/(%X40000000/__CEIL)
$
$ RETURN
```

8.5   What does the MCR command do?

The MCR is an artifact of RSX compatibility mode, the operating system
from which OpenVMS is descended. MCR is the Monitor Console Routine,
and the command is intended to activate RSX compatibility mode utilities.
When used on OpenVMS, the command is most commonly used to run
the specified image and—because the tool detects the image is not a
compatibility-mode image—it acts as a form of RUN command with
the default file specification of SYS$SYSTEM:.EXE. MCR passes any
(optional) command line arguments in a fashion similar to a foreign
command. In other words:

```
$ MCR FOO BAR
```

is equivalent to:

```
 $ FOO :== $FOO
 $ FOO BAR
```

MCR is not documented. Use of a foreign command or the DCL$PATH
mechanism is prefered. For details on this, see Section 8.1.

8.6   How do I change the OpenVMS system prompt?

You can use the SET PROMPT command for this purpose. SET PROMPT
sets the DCL prompt to the specified string.

When you want to display variable information, you will need to establish
a tie-in that provides the information to the SET PROMPT command as
required.

If you wish to display the default directory for instance, you will have
to establish a tie between the SET DEFAULT command and the SET
PROMPT commands, as there is no direct way to get the default directory
as the DCL prompt. You can easily acquire or create a set of DCL
command procedures that perform the SET DEFAULT and SET PROMPT
for you. These DCL command procedures often use a command such as:

```
$ set prompt='f$environment("default")'
```

More advanced users could implement a system service or other intercept,
and use these tools to intercept the directory change and reset the
prompt accordingly. (This approach likely involves some kernel-mode
programming, and requires write access to various undocumented
OpenVMS data structures.)

There are related tools available from various sources, including the following web sites:

- ftp://ftp.hhs.dk/pub/vms/setpmt/

- ftp://ftp.tmesis.com/sys_service_hook.src

- James F. Duff has also made available a Macro32 tool known as TIME_PROMPT, a tool that sets the prompt to the current system time.

- Many folks have contributed DCL procedures to perform this task. Visit the newsgroup archives for information and examples.

8.7 Can I do DECnet task-to-task communication with DCL?

Yes, you can do this with DCL.

The OpenVMS DECnet documentation shows various simple examples using the task object and the TYPE command to trigger the execution of a DCL command procedure on a remote node. An example DCL command procedure that is rather more advanced than using the TYPE command as a trigger is included in the Ask The Wizard area:

- http://www.openvms.compaq.com/wizard/

- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

DCL does not include support asynchronous I/O, thus a predetermined protocol or a predetermined "turn-around" command sequence must be implemented in order to avoid protocol deadlocks—cases where both tasks are trying to write or both tasks are trying to read. The task that is writing messages to the network must write (or write and read) a predetermined sequence of messages, or it must write a message that tells the reader that it can now start writing messages. (This is the essence of a basic half-duplex network protocol scheme.)

8.8 How can I get the width setting of a terminal?

```
$ width = f$getdvi(terminal,"DEVBUFSIZ")
```

8.9 How can I substitute symbols in a PIPE?

Use DCL ampersand substitution, and not apostrophe substitution.

```
$ pipe show system | search sys$input opcom | (read sys$input pid ;
    pid=f$element(0," ",pid) ; define/system opcom_pid &pid)
$ show log opcom_pid
    "OPCOM_PID" = "0000020B" (LNM$SYSTEM_TABLE)
```

8.10 Use of RUN/DETACH, LOGINOUT, and logical names?

With a command to create a detached process such as:

```
$ RUN/DETACHED SYS$SYSTEM:LOGINOUT /INPUT=TEMP_INPUT.COM
```

If you are trying to use a logical name as the /INPUT, /OUTPUT or
/ERROR on a RUN/DETACH command, then you must translate the
logical name specifications to physical references before passing them, or
the definitions must reside in a logical name table that is visible to the
newly-created process.

Also note that LOGINOUT only creates the SYS$LOGIN, SYS$LOGIN_
DEVICE, and SYS$SCRATCH logical names if it is processing a login
that is based on the contents of a SYSUAF record—without access
to the associated SYSUAF record, this information is not available to
LOGINOUT. (If you want to see these particular logical names created,
then please specify the /AUTHORIZE qualifier on the RUN/DETACHED
command.)

If you do not specify LOGINOUT as the image, then there is no easy
way to get these logical names. Also, any logical names that are used in
the target image file specification must also be in a logical name table
accessable (by default) by the newly-created detached process. Shared
tables include the group (if the process is in the same UIC group) and
the system table. (If the target process is to be in another UIC group, a
suitablly privileged user or application can create the necessary logical
name(s) directly in the other group logical name table.)

When in doubt, create a short DCL command file as input, and use a
SHOW LOGICAL and similar commands to examine the context. (And
use physical device and directory references on the RUN/DETACH of the
LOGINOUT image, when specifying this command file as /INPUT.) Also
remember to check both security auditing and system accounting when
troubleshooting problems with the RUN/DETACH.

Also see Section 8.1.

8–6

8.11 How to use escape and control characters in DCL?

To write a message and then the bell character, use:

```
$ bell[0,7] = 7
$ write sys$output "Hello''bell'"
```

To write blinking text, use:

```
$ esc[0,7] = 27
$ text = "Blinking Text"
$ write sys$output "''esc'[5m''text'''esc'[m"
```

Also see sections Section 11.7, Section 12.1.

9 Files

9.1 How can I undelete a file?

OpenVMS doesn't have an "undelete" function. However, if you are quick to write-protect the disk or if you can guarantee that no new files get created or existing files extended, your data is still on the disk and it may be possible to retrieve it. The FLORIAN tool available from various websites can potentially recover the file, see question Section 13.1 for pointers. Other alternatives here include the DFU tool, available on the OpenVMS Freeware CD-ROM distribution.

If you are setting up a user environment for yourself or for others, it is quite easy to use DCL to intercept the DELETE command, using a symbol:

```
$ DEL*ETE :== @SYS$LOGIN:MYDELETE.COM
```

The DELETE symbol will cause the procedure to be invoked whenever the user enters the DELETE command, and it can copy the file(s) to a "trashcan" subdirectory before issuing a "real" DELETE on the files. Other procedures can retrieve the file(s) from the "trashcan" subdirectory, and can (and should) clean out the "trashcan" as appropriate. (Realize that this DELETE symbol can interfere with DELETE/GLOBAL and other similar DCL commands.)

9.2 Why does SHOW QUOTA give a different answer than DIR/SIZE?

DIRECTORY/SIZE doesn't take into account the size of file headers which are charged to your quota. Also, unless you use DIRECTORY/SIZE:ALL, you will see only the "used" size of the file, not the allocated size which is what gets charged against your quota. Also, you may have files in other directories.

```
$ DIRECTORY/SIZE=ALL/GRAND [username...]
Grand total of D1 directories, F1 files, B1/B2 blocks.
$ DIRECTORY/SIZZ=ALL/GRAND [-]username.DIR
Grand total of 1 directory, 1 file, B3/B4 blocks.
$ SHOW QUOTA
User [username] has B5 blocks used, B6 available
of B7 authorized and permitted overdraft of B8 blocks on disk
```

If the user has no files in other directories and all file-headers are only 1 block, then the following should apply:

```
B5=B2+B4+F1+1
```

If the diskquota has drifted out of synchronization, then the system-manager can force a quota rebuild—due to various factors, the quota file can potentially drift from the actual use over time, and a periodic rebuild can be performed at appropriate intervals.

Also be aware that the DIRECTORY/SIZE command can report larger values than might otherwise be expected when used to evaluate files and/or directories that are alias links—such as the system roots on OpenVMS system disks—as the command reports a total that is cumulative over all of the files and directories examined, without regard for which ones might be alias entries and which are not. (In other words, a DIRECTORY/SIZE of an entire OpenVMS system disk will report a disk useage value larger than the (usually more accurate) value reported by the SHOW DEVICE command. This as a result of the alias entries linking each SYS$SYSDEVICE:[SYSCOMMON]SYS*.DIR directory file and the SYS$SYSDEVICE:[000000]VMS$COMMON.DIR file together.)

9.3 How do I make sure that my data is safely written to disk?

If your application must absolutely guarantee that data is available, no matter what, there's really no substitute for RMS Journaling and host- or controller-based shadowing. However, you can achieve a good degree of data integrity by issuing a SYS$FLUSH RMS call at appropriate times (if you're using RMS, that is.) If you're using a high-level language's I/O system, check that language's documentation to see if you can access the RMS control blocks for the open file. In C you can use fflush followed by fsync.

For details on disk bad block handling on MSCP and on SCSI disk devices, please see Ask The Wizard (ATW) topic (6926).

- http://www.openvms.compaq.com/wizard/
- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

9.4   What are the limits on file specifications and directories?

A file specification has an aggregate maximum size of 255 characters at present. The node and device specification may be up to 255 characters each - file name and file types may be up to 39 characters each. File versions are from 1 through 32767, though 0 (latest version), -0 (oldest version) and -n (n'th previous version) can be used in most contexts. A file specification may not have more than 8 directories and subdirectories - while it is possible to create subdirectories of greater depth, accessing them is problematic in most cases and this should be avoided.

Application developers should use OpenVMS-supplied routines for parsing file specifications - this ensures that changes in what is allowable will not tend to break your application. Consider that various parts of the file specification may contain quoted strings with embedded spaces and other punctuation! Some routines of interest are SYS$FILESCAN, SYS$PARSE and LIB$TRIM_FILESPEC. For further information, see the OpenVMS Guide to File Applications.

Performance of larger directory files improves (greatly) with OpenVMS V7.2 and later—operations on directory files of 128 blocks and larger were rather slower on earlier OpenVMS releases due to the smaller size of the directory cache and due to the directory I/O processing logic.

For fastest directory deletions, consider a reverse deletion—delete from the last file in the directory to the first. This reversal speeds the deletion operation by avoiding unnecessary directory I/O operations as the files are deleted. Tools such as the Freeware DFU can be used for this purpose, as can various available reverse-DELETE DCL command procedures.

9.5   What is the largest disk volume size OpenVMS can access?

One Terabyte (TB; 2**31 blocks of 2**9 bytes; 0x07FFFFFFF blocks). 255 volumes in a volume set. The largest contiguous allocation possible for any particular file is 0x03FFFFFFF blocks.

Prior to the release of V6.0, the OpenVMS file system was limited to disk volumes of 8.38 GB (2**24 blocks, 16777216 blocks) or less.

On some systems, there are restrictions in the console program that limit the size of the OpenVMS system disk. Note that data disks are not affected by console program limits. For example, all members of the VAXstation 3100 series are limited to a system disk to 1.073 GB or less due to the console, though larger data disks are possible.

Some SCSI disks with capacities larger than 8.58 gigabytes (GB) will require the use of an OpenVMS ECO kit (eg: ALPSCSI04_062 or later) for new SCSI device drivers. Failure to use this ECO can cause "rounding errors" on the SCSI disk device capacity—OpenVMS will not use nor display the full capacity of the drive—and "%sysinit-e-error mounting system device status equals 000008C4" (8C4 -> "%SYSTEM-?-FILESTRUCT, unsupported file structure level") errors during bootstrap. (One workaround for the bootstrap when the bitmap is located far into the disk is the use of INIT/INDEX=BEGIN.) The problem here involves the particular extensions and fields used for larger capacity disks within the SCSI specifications and within the various intepretations of same.

For IDE disk drives:

- Versions of SYS$DQDRIVER *BEFORE* X-15 topped out at 8.455 GB.

  Fixed drivers (>="X-15") were shipped in:

  - OpenVMS Alpha V7.2-1, and later

  - V7.2 UPDATE V1.0 ECO, and later

  - V7.1-2 UPDATE V1.0 ECO, and later

  - V7.1-2 UPDATE V3.0 ECO, and later

- The newer SYS$DQDRIVER driver operates to disks up to 33 GB without (known) problems, and effectively works with rather larger disks (up to circa 137 GB) but is known to report an incorrect number of "cylinders" with disks above 33 GB.

See Section 14.21 for additional IDE SYS$DQDRIVER information.

Be aware that a known restriction in certain older versions of the Alpha SRM Console prevents booting most IDE drives larger than 8.455 GB, depending on exactly where the various files are located on the volume. Updated SRM consoles for systems with SRM and IDE drive support are (will be) available. (OpenVMS Engineering has successfully bootstrapped 20GB IDE disks using the appropriate SRM console version.)

NOTE: All IDE-related disk sizes listed in this section are stated in units of "disk (base ten) gigabytes" (1 GB = 10^9 bytes) and NOT in units of "software (base two) gigabytes" (1 GB = 2^{30} (1073741824.) bytes. See Section 14.44.

Be aware that larger disks that are using an extension of SCSI-2— disks that are using a mode page field that the SCSI-2 specifications normally reserved for tape devices—to permit a larger disk volume size will require a SCSI driver update for OpenVMS, and this change is part of V7.1-2 and later, and also part of ALPSCSI07_062 and later. (These larger disks disks will typically report a DRVERR, or will see the volume size "rounded down".) SCSI disks larger than 16777216 blocks cira 8.455 GB (base ten); 8GB (base two) require this ECO, or require the use of OpenVMS Alpha V7.1-2 or later.

Applications written in C can be limited to file sizes of two gigabytes and less, as a result of the use of longword values within C file operations, and specifically off_t. This restriction is lifted in OpenVMS V7.3-1 and later, and with the application of the C ECO kits available for specific earlier releases. The use of a longword for off_t restricts applications using native C I/O to file sizes of two gigabytes or less, or these applications must use native RMS or XQP calls for specific operations.

Also see Section 14.27, Section 14.44.

9.6 What is the maximum file size, and the RMS record size limit?

RMS can store individual files of a size up to the maximum supported volume size. Under OpenVMS V6.0 and later, the volume size and the RMS maximum file size limit is 2**31 * 512 bytes—one terabyte (1 TB).

"Use a volume set to provide a large, homogeneous public file space. You must use a volume set to create files that are larger than a single physical disk volume. (The file system attempts to balance the load on the volume sets, for example, by creating new files on the volume that is the least full at the time.)"

"You can add volumes to an existing volume set at any time. The maximum number of volumes in a volume set is 255."

The RMS formats—sequential, relative, and indexed—are limited by the one terabyte maximum volume size. RMS relative files are further limited to a number of records that will fit in 32 bits—4 billion records. Sequential and indexed formats do not have a record limit.

Also see Section 10.13, Section 14.44.

9.7 How do I write recordable CD media (CD-R) on OpenVMS?

At its simplest: get LDDRIVER from the Freeware. Get CDRECORD or CDWRITE. (CDRECORD and CDWRITE are freely available, though versions are not on the Freeware V5.0 distribition; the URLs are referenced later in this section). Build the contents of the disk on the LD device partition. Then use the CDRECORD or CDWRITE tool to record the contents of the LD partition directly onto the CD-R or CD-RW media.

While folks have had success getting PC-based CD-R/CD-RW tools to work with OpenVMS partitions, it is far easier and more reliable to use the OpenVMS-based versions of the tools.

More details: Creation of CD-ROMs using CD-Recordable media (CD-R) under OpenVMS typically involves one of two approaches: the use of the optional CD-R ('Scribe') capabilities available for the InfoServer or other "offline" hardware packages (PC-based packages will be included in this), or the use of a host-based package such as the CDWRITE13_VMS utility, an OpenVMS port of a Linux tool.

OpenVMS presently has no integrated support for recording CD-R media, though this is expected to change in a release after OpenVMS Alpha V7.3.

OpenVMS can read both ODS2 and ISO9960 format CD-ROMs.

InfoServer hardware configurations are no longer available from HP, but may potentially be acquired through other means.

The CDWRITE13_VMS package is one example of a host-based package that can be used to create CD-R media. The contact for CDWRITE13_VMS is Dr. Eberhard Heuser-Hofmann. One website that discusses this package is located at:

```
http://www.geocities.com/SiliconValley/Lakes/9999/vmscdwri.html
```

Additional information is available at the following sites:

- http://www.djesys.com/vms/cdrom.html
- http://www.cd-info.com/CDIC/Technology/CD-R/vms.html
- http://www.faqs.org/faqs/cdrom/cd-recordable/part1/preamble.html
- http://www.tmesis.com/CDrom/
- http://www.tditx.com/~odsiso/

U.S. Design offers a package that includes the tools necessary to create a CD or DVD-R with either 9660 or ODS-2 format, for standalone CD-R or DVD-R drives, for recent OpenVMS versions. Details are available at:

* http://www.usdesign.com/

9.8 What I/O transfer size limits exist in OpenVMS?

The maximum transfer size is an attribute of the particular I/O device, controller and driver combination; there is no inherent limit imposed by OpenVMS (other than the fact that, today, byte counts and LBNs are generally limited to 32 bits).

The maximum size of a device I/O request is limited by the value in UCB$L_MAXBCNT, which is set by the device driver based on various factors. (Also check the setting of the MAXBUF system parameter for buffered I/O transfers, and check the process quotas.)

Currently, SCSI drivers limit I/O transfers to FE00(16) bytes, 65024 bytes (decimal). The reasons for this transfer size limitation are largely historical. Similarly, DSSI devices are limited to the same value, this for hardware-specific reasons. Transfers to HSC and HSJ device controllers via the CI are limited to 1,048,576 bytes. Client MSCP-served devices are limited to 65535 bytes—to help ensure that the I/O fragmentation processing happens on the client and not on the server system.

Parts of the OpenVMS I/O subsystem are optimized for data transfers less than 64KB, because (obviously) most I/O operations are (substantially) less than that. OpenVMS can handle larger transfers, if the driver and the device can handle it.

Also see Section 9.4, Section 9.5.

9.9 Can I use ODBC to connect to OpenVMS database files?

Yes, you can use various available third-party packages that permit remote ODBC clients to access RMS files and various commercial databases via the network.

For RMS, consider acquiring one of the packages available from EasySoft, Attunity Connect (formerly known as ISG Navigator), Oracle (DB Integrator), SolutionsIQ, and Synergex.

For specific commercial databases (other than RMS, of course), contact the database vendor directly for assistance.

10 OpenVMS Programming Information

10.1 Can I have an example of calling...?

Please use the available SEARCH command on OpenVMS, and please remember to search the available resources, including the support databases and the newsgroup archives. Please also realize that most OpenVMS system services use similar calling sequences, meaning that an example of calling sys$getjpi can be used as an example for sys$getsyi and sys$getdvi. Students: please do not expect folks to write your homework for you. As for search resources:

- SEARCH SYS$EXAMPLES:*.* target

- SEARCH TCPIP$EXAMPLES:*.* target

- http://askq.compaq.com/

- http://www.openvms.compaq.com/freeware/

- http://www.google.com/

OpenVMS programming documentation, including the numerous example programs found in recent versions of the *OpenVMS Programming Concepts* manual, is available:

- http://www.openvms.compaq.com:8000/

- http://www.openvms.compaq.com/doc/

- http://www.openvms.compaq.com/commercial/

As for details of argument passing, most OpenVMS system services and RTL routines pass string arguments by descriptor. Languages which support native string data types create descriptors automatically; those which do not (eg., C) require that you set them up explicitly. For further details on using descriptors and particularly for using descriptors from C, please see Section 10.12.

There is extensive information available on how to call OpenVMS system services and OpenVMS Run-Time Library routines, including examples in numerous languages. Among the best available references are:

- Your language's User Manual

- *OpenVMS Programming Environment Manual*
- *OpenVMS Programming Concepts Manual*
- *OpenVMS Programming Interfaces: Calling a System Routine*
- *OpenVMS Calling Standard*

In addition, if you are a subscriber to the HP Software Information Network (available to those with a software support contract), the support database contains hundreds of worked examples of calling system services and RTL routines, including the one that seems to trip up almost everyone, SMG$CREATE_MENU.

Arne Vajhøj has put together a collection of OpenVMS example programs. It can be found at:

- ftp://ftp.hhs.dk/pub/vms/

Additional information and examples for OpenVMS are available via:

- http://www.openvms.compaq.com/wizard/
- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9. and via:

- http://www.openvms.compaq.com/freeware/

10.2 How do I get the arguments from the command line?

If you're writing a program and want to accept arguments from a foreign command, you can use LIB$GET_FOREIGN to get the command line and parse it yourself, or if you're programming in C, use the normal argc/argv method.

To write an application which uses the normal DCL verb/qualifier/parameter syntax for invocation, see the description of the CLI$ routines in the OpenVMS Callable Utility Routines Reference Manual.

It is possible to write an application which can be used both ways; if a DCL verb isn't used to invoke the image, the application parses the command line itself. One way to do this is to call CLI$GET_VALUE for a required parameter. If it is not present (or you get an error), call LIB$GET_FOREIGN to get the command line and do the manual parse.

See also reference>(DCL1).

10.3 How do I get a formatted error message in a variable?

Use the SYS$PUTMSG system service with an action routine that stores the message line(s) in the variable of your choice. Be sure the action routine returns a "false" (low bit clear) function value so that SYS$PUTMSG doesn't then try to display the message (unless you want it to.) See the description of $PUTMSG in the System Services Reference Manual for an example of using an action routine.

10.4 How do I link against SYS$SYSTEM:SYS.STB on an Alpha system?

LINK/SYSEXE is the OpenVMS Alpha equivalent of linking against SYS.STB. This links against the base image: SYS$BASE_IMAGE.EXE

Also see Section 10.10, particularly for pointers to the details on shareable images and shareable image creation.

10.5 How do I do a SET DEFAULT from inside a program?

The problem is that SYS$SETDDIR only changes the default directory - NOT the default disk. The default disk is determined by the logical SYS$DISK. If you want to change the default disk within a program, then call LIB$SET_LOGICAL to change the logical SYS$DISK. You will need to call both LIB$SET_LOGICAL and SYS$SETDDIR to change both default disk and the default directory!

10.6 How do I turn my Fortran COMMON into a shareable image on Alpha?

You need to add SYMBOL_VECTOR=(<common-name>=PSECT) to your options file. On OpenVMS VAX all OVR/REL/GBL psects were automatically exported into the shareable image's Global Symbol Table. On OpenVMS Alpha you have to tell the linker that you want this done by means of the PSECT keyword in the SYMBOL_VECTOR options file statement.

This has several advantages over OpenVMS VAX. First, you don't have to worry about the address of the psect when you try to create a new, upwardly compatible version of the shareable image. Second, you can control which psects, if any, are made visible outside the shareable image.

By default, COMMON PSECTs in HP Fortran for OpenVMS Alpha (as well as most other OpenVMS Alpha compilers) are NOSHR. On VAX, the default was SHR which required you to change the attribute to NOSHR

if you wanted your COMMON to be in a shareable image but not write-shared by all processes on the system. If you do want write-sharing, use:

```
CDEC$ PSECT common-name=SHR
```

in the Fortran source code (the CDEC$ must be begin in column 1) or a linker options file PSECT_ATTR statement to set the COMMON PSECT attribute to SHR.

For further information, see the Linker manual.

10.7   How do I convert between IEEE and VAX floating data?

In OpenVMS V6.1 and later, the routine CVT$CONVERT_FLOAT is documented in the LIB$ Run-Time Library Reference Manual, and can perform floating point conversions between any two of the following floating datatypes: VAX (F,D,G,H), little-endian IEEE (single, double, quad), big-endian IEEE (single, double, quad), CRAY and IBM System\370, etc.

HP Fortran (all OpenVMS platforms) has a feature which will perform automatic conversion of unformatted data during input or output. See the HP Fortran documentation for information on "non-native data in I/O" and the CONVERT= OPEN statement keyword.

There are floating-point conversion source code packages available for various platforms.

For further floating-point related information, see:

- http://www.hhs.dk/anonymous/pub/vms/collection/ieee.zip

10.8   How do I get the argument count in a Fortran routine?

On VAX, many programmers would use a MACRO routine which accessed the AP register of the caller to get the address of the argument list and hence the argument count. This was not guaranteed to work on VAX, but usually did. However, it doesn't work at all on OpenVMS Alpha, as there is no AP register. On Alpha systems, you must use a language's built-in function to retrieve the argument count, if any. In Fortran this is IARGCOUNT, which is also available in DEC Fortran on OpenVMS VAX.

Note that omitting arguments to Fortran routines is non-standard and is unsupported. It will work in many cases - read the DEC Fortran release notes for additional information.

10–4

10.9   How do I get a unique system ID for licensing purposes?

Many software developers desire to use a unique hardware ID to "lock" a given copy of their product to a specific system. Most VAX and Alpha systems do not have a unique hardware-set "system ID" that can be used for this purpose. HP OpenVMS products do not use hardware IDs in the licensing methods, as many users consider a hardware-based licensing scheme to be negative attribute when considering software purchases.

HP OpenVMS uses a software-based system called the License Management Facility (LMF). This provides for software keys (Product Authorization Keys or PAKS) which support capacity and user-based license checking. HP offers an LMF PAK Generator to CSA members—see Section 2.12.

For information on licensing, please see Section 12.5.

However, if a hardware-based method is required, the most common method is based on an Ethernet adaptor hardware address. Sample source code for implementing this is available at:

- http://www.openvms.compaq.com/wizard/

- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

10.10   What is an executable, shareable, system or UWSS image?

Executable code in OpenVMS typically resides in an image—an image is a file—the file extension is typically .EXE—that contains this code. Common types of images include executable images, shareable images, system images, and protected (UWSS) images.

Executable images are programs that can be directly executed. These images can grant enhanced privileges, with an INSTALL of the image with /PRIVILEGE, or can grant enhanced access with the specification of a subsystem identifier on the ACL associated with the image.

Shareable images contain code executed indirectly, these images are referenced from executable images and/or from other shareable images. These images can not grant enhanced privileges, even with the use of INSTALL with /PRIVILEGE or a subsystem identifier. These shareable images can be dynamically activated (a LINK that occurs at run-time) via the LIB$FIND_IMAGE_SYMBOL run-time library (RTL) routine. (See 'protected images' for information on 'privileged shareable images'.)

System images are intended to run directly on the VAX or Alpha hardware—these are normally used for the kernel code that comprises an operating system.

Protected images—also refered to as User-Written System Services (UWSS), or as privileged shareable images—are similiar in some ways to a standard shareable images, but these images include a 'change mode' handler, and execute in an 'inner' processor mode (privileged mode; executive or kernel), and code executing in inner modes has implicit SETPRV privilege. Must be INSTALLed with /PROTECT. Note that inner-mode code has restrictions around calling library routines, around calling various system services, and around calling code located in other protected or shareable images.

Loadable images and device drivers are images that can be used to add code into the OpenVMS kernel. Pseudo-device drivers are a particularly convenient way to add executable code, with associated driver-defined data structures, into the kernel. The pseudo-device driver includes the UCB and DDB data structures, <>(UCB) and a calling interface with support for both privileged and unprivileged access to the driver code via sys$qio[w] calls. <>(sys$qio)

A cookbook approach to creating OpenVMS shareable images is available at the URL:

- http://www.openvms.compaq.com/wizard/

- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

10.11   How do I do a file copy from a program?

There are several options available for copying files from within a program. Obvious choices include using lib$spawn(), system(), sys$sndjbc() or sys$creprc() to invoke a DCL COPY command. Other common alternatives include using the callable convert routines and the BACKUP application programming interface (V7.1 and later). <>(lib$spawn) <>(COPY)

10.12   What is a descriptor?

A descriptor is a data structure that describes a string or an array. Each
descriptor contains information that describes the type of the data being
referenced, the size of the data, and the address of the data. It also
includes a description of the storage used for the data, typically static or
dynamic. Descriptors are passed by reference.

The following are examples of creating and using descriptors in C,
with the use of the angle brackets normally expected by the C include
statements deliberately altered in deference to HTML:

```
#include {descrip.h}
#include {lib$routines.h}
#include {stsdef.h}
int RetStat;
char TxtBuf[TXTSIZ]
struct dsc$descriptor StaticDsc =
  { 0, DSC$K_DTYPE_T, DSC$K_CLASS_S, NULL };
struct dsc$descriptor DynDsc =
  { 0, DSC$K_DTYPE_T, DSC$K_CLASS_D, NULL };
int DynDscLen = 255;
$DESCRIPTOR( ConstDsc, "This is a string" );

/* finish setting up a static descriptor */
StaticDsc.dsc$w_length    = TXTSIZ;
StaticDsc.dsc$a_pointer    = (void *) TxtBuf;

/* finish setting up a dynamic descriptor */
RetStat = lib$sget1_dd( &DynDscLen, &DynDsc );
if ( !$VMS_STATUS_SUCCESS( RetStat ) )
  return RetStat;

/* release the dynamic storage */
RetStat = lib$sfree1_dd( &DynDsc );
if (!$VMS_STATUS_SUCCESS( RetStat ))
  return RetStat;
```

<>(lib$sget1_dd)

Static descriptors reference storage entirely under application program
control, and the contents of the descriptor data structure can be modified
as required (by the application). OpenVMS routines do not modify the
contents of a static descriptor, nor do they alter the address or length
values stored in the static descriptor. (The term "static" refers to the
descriptor data structure, and not necessarily to the storage referenced by
the descriptor.)

Dynamic descriptors reference storage under the control of the run-time library, and the contents of a dynamic descriptor data structure—once initialized—can only be modified under control of run-time library routines. The dynamic storage referenced by the dynamic descriptor is allocated and maintained by the run-time library routines. Various OpenVMS routines do alter the contents of the descriptor data structure, changing the value for the amount and the address of the storage associated with the dynamic descriptor, as required. Routines can obviously access and alter the contents of the storage referenced by the descriptor.

OpenVMS languages that include support for strings or arrays are expected to use descriptors for the particular structure. Most OpenVMS languages, such as Fortran and BASIC, use descriptors entirely transparently. Some, like DEC C, require the programmer to explicitly create and maintain the descriptor.

For further information on string descriptors, see the *OpenVMS Programming Concepts* manual, part of the OpenVMS documentation set.

Fortran defaults to passing integers by reference and characters by descriptor. The following sites discuss mixing Fortran and C source code in the same application:

- http://www.hhs.dk/anonymous/pub/vms/misc/FORTRAN_C_CALL.COM

- ftp://ftp.hhs.dk/pub/vms/misc/FORTRAN_C_CALL.COM

10.13 How many bytes are in a disk block?

A disk block is the minimum unit of disk storage allocation in OpenVMS.

Under OpenVMS VAX and OpenVMS Alpha, the disk volume block size is consistent, with each block containing 512 bytes.

The minimum disk allocation granularity actually permissible (in the ODS-2 and ODS-5 volume structures commonly used on OpenVMS) is determined on a per-volume basis, and is typically based on a combination of the total number blocks on the disk volume and the total size of the volume storage bitmap. The allocation granularity is known as the volume cluster factor—the cluster factor is the number of blocks in a disk cluster, and it is the smallest number of blocks that can be allocated on a particular disk volume.

Prior to OpenVMS V7.2, the maximum permissible size of the bitmap requires larger cluster factors as volume sizes increase. Starting with V7.2, the bitmap can be larger, and cluster factors as small as one block can be used.

The number of bytes in a file can be determined by multiplying the number of blocks allocated for the file times the number of bytes in a block. For sequential files (only), the FFB (XAB$W_FFB, in the File Header XAB) value can be used to find out how much of the last (XAB$L_EBK) block is used. FFB and EBK are meaningful only for sequential files, and only in a limited context—partial block allocations are not permitted. For other file formats, the EOF marker is not meaningful.

Disk allocations always occur only in units of the cluster factors, which can be from one block up to (potentially) clusters of eighteen blocks or more, depending on the volume cluster factor. (OpenVMS V7.2 and later optionally provide for a cluster factor of one up to volumes of approximately 137 gigabytes.)

OpenVMS assumes that the device driver and the underlying storage device will present the file system with addressable units of storage of 512 bytes in size, or the appearance of same. Various third-party CD-ROM devices, for instance, support only 2048 byte blocks, and such devices are incompatible with the standard OpenVMS device drivers.

To determine the number of bytes required for a file from DCL, one option uses the f$file_attributes item EOF, multiplied by the size of a block in bytes (512). This does not account for the unused space in the last block of a sequential file, but it also does not have to differentiate sequential files from other files.

10.14   How many bytes are in a memory page?

A memory page is the minimum unit of memory allocation in OpenVMS. With OpenVMS VAX, the memory page size matches the disk block size: it is always 512 bytes.

With OpenVMS Alpha, the memory page size is variable, and it can range from 8192 bytes (8 kilobytes) up to 64 kilobytes. The current system page size can be determined using the sys$getsyi or f$getsyi PAGE_SIZE item. Programs with hardcoded constants for the memory page size (or page alignment) should always assume a page size of 64 kilobytes.

On OpenVMS Alpha, a 512 byte area of memory—equivilent in size to an OpenVMS VAX memory page—is refered to as a pagelet.

10.15   How do I create a process under another username?

Many server processes can operate within the context of the target user using privileges, using calls such as sys$chkpro and (more commonly in this context) sys$check_access as needed to determine if access would be permitted for the specified user within the current security model.

With OpenVMS V6.2 and later, the persona system services (SYS$PERSONA_*) can be used to assume the persona of the specified user—these allow the server to operate as the specified user, in a controlled fashion. The persona services can be used as a "wrapper" around a sys$creprc process creation call, as well—this will create a seperate process entirely under the assumed persona.

Information on the persona system services is included in the OpenVMS V6.2 new features documentation, and in the OpenVMS V7.1 and later system services documentation. These system services exist and are supported in OpenVMS V6.2 and later releases.

Typical mechanisms for creating a process under another username include:

- personna services around a sys$creprc call. See above.

- via DECnet task-to-task, using explicit specification of username and password, or using a DECnet proxy. This creates a network-mode job under the target user. The network-mode job might do little more than a RUN/DETACH of an image passed in via task-to-task—task-to-task communications are fully available using strictly DCL-to-DCL processing, or using a compiled language and DCL, etc.)

- SUBMIT/USER, or the username argument on the sys$sndjbc call. This creates a batch-mode job under the specified username. The batch-mode job might do little more than a RUN/DETACH of an image passed in via a parameter.

- the UIC argument on the sys$creprc call. This mimics the UIC of the target user, and is certainly not the prefered mechanism for this task.

- Via pseudo-terminals...

There are likely a few other mechanisms around... There are various tools available from DECUS and other sources that allow various forms of user impersonation, as well. These tools will require version-dependent kernel code and enhanced privileges for some of (or all of) their operations.

10–10

10.16   Why do lib$spawn, lib$set_symbol fail in detached processes?

The processing within run-time library (RTL) calls such as lib$attach, lib$disable_ctrl, lib$do_command, lib$enable_ctrl, libget_symbol, librun_program, libset_symbol, libset_logical, and lib$spawn, is dependent on and requires the presence of a command language interpreter (CLI), such as DCL. Without a CLI present in the current process, these calls will fail with a "NOCLI, no CLI present to perform function" error.

Detached processes typically do not have a CLI present.

In place of lib$spawn, sys$creprc can often be used. The context of the parent process (symbols and logical names) will not be propogated into the subprocess when sys$creprc is used, though when there is no CLI present in the process this (lack of) propogation is moot.

To create a detached process with a CLI, you must specify LOGINOUT as the target image as discussed elsewhere in the FAQ, or only use these calls (and any other calls requiring a CLI) from images that are running in an "interactive", "batch", or "other" mode process.

Also note that the lib$spawn and the C system call will fail in a CAPTIVE login environment. The lib$spawn call can be gotten to work in this environment with the specification of the TRUSTED flag.

10.17   Where can I obtain Bliss, and the libraries and supporting files?

The Bliss language compilers and documentation are available on the OpenVMS Freeware distributions.

Bliss language source code that contains the following statement:

```
LIBRARY 'SYS$LIBRARY:STARLET.L32';
```

or similar requires the presence of the Bliss libraries. These libraries are created on the target system using the Bliss require files, and are built using the following Bliss commands:

STARLET.L32 contains the public interfaces to OpenVMS:

```
$ BLISS /LIBRARY=SYS$COMMON:[SYSLIB]STARLET.L32 -
    SYS$LIBRARY:STARLET.REQ
```

LIB.L32 contains both the public and private interfaces to OpenVMS:

```
$ BLISS /LIBRARY=SYS$COMMON:[SYSLIB]LIB.L32 -
    SYS$LIBRARY:LIB.REQ+SYS$LIBRARY:STARLET.REQ
```

The equivilent files for Bliss64 are created with:

```
$ BLISS/A64/LIBRARY==SYS$COMMON:[SYSLIB]LIB.L64 -
     SYS$LIBRARY:LIB.R64+STARLET.REQ+STARLET.R64
$ BLISS/A64/LIBRARY==SYS$COMMON:[SYSLIB]STARLET.L64 -
     SYS$LIBRARY:STARLET.R64
```

Some Bliss code may also require the OpenVMS VAX architecture flags.
The following is the equivilent of the Alpha ARCH_DEFS.BLI module:

```
!
! This is the OpenVMS VAX version of ARCH_DEFS.BLI, and
! contains the architectural definitions for conditionally
! compiling OpenVMS Bliss sources for use on VAX systems.
!
MACRO VAXPAGE = 1%;
MACRO BIGPAGE = 0%;

MACRO VAX =                        ! = 1 if compiled BLISS/VAX
        %BLISS(BLISS32V)%;         ! = 0 if not compiled BLISS/VAX

MACRO EVAX =                       ! = 1 if compiled BLISS/E*
!
! A more appropriate definition can only be used with versions
! of the Bliss compilers that understand the 32E/64E flags.
!        %BLISS(BLISS32E) OR %BLISS(BLISS64E)%; ! = 0 if compiled /VAX
         NOT %BLISS(BLISS32V)%;  ! = 0 if compiled /VAX

MACRO ADDRESSBITS =
        %BPADDR%;                  ! = 32 or 64 based on compiler used
```

10.18 How can I open a file for shared access?

When creating a file, it is often useful to allow other applications and
utilities—such as TYPE—to share read access to the file. This permits
you to examine the contents of a log file, for instance.

A C source example that demonstrates how to do this is available in topic
(2867) in the OpenVMS Ask The Wizard area:

- http://www.openvms.compaq.com/wizard/

- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

Depending on the environment, you may need to use C calls such as fsync
and fflush, and—in specific cases—the setvbuf(_IONBF) call.

10.19   How can I have common sources for messages, constants?

Use the GNM tools on the OpenVMS Freeware to have common sources
for MSG (message) files and SDML (Document) documentation files. Use
the DOCUMENT command to convert the SDML documentation into the
necessary formats (Text, Postscript, HTML, etc). Use the MESSAGE/SDL
tool (latent in OpenVMS) to create an SDL file based on the messages.
Then use the SDL tool (available on the OpenVMS Freeware) to convert
the SDL file into language-specific definitions. (There is also a converter
around to convert SDL into SDML, if you want to get pictures of the data
structures for your documentation.)

10.20   How do I activate the OpenVMS Debugger from an application?

```
#include {lib$routines.h}
#include {ssdef.h}
#include {string.h}

main()
    {
    char ascic_debug_commands[128];
    char *dbgcmd = "*show calls;go;exit";

    strcpy( ascic_debug_commands, dbgcmd );
    ascic_debug_commands[0] = (char) strlen( dbgcmd ) - 1;

    lib$signal(SS$_DEBUG,1,ascic_debug_commands);

    return 1;
    }
```

10.21   Dealing with Endian-ness?

OpenVMS on VAX, OpenVMS on Alpha and OpenVMS on Intel IA-
64 platforms (as well as all Microsoft Windows implementations and
platforms) all support and all use the little-endian byte ordering. Certain
Alpha microprocessors and certain Intel Itanium processors can be
configured to operate in big-endian and potentially in bi-endian mode.
HP-UX typically operates big-endian.

With little-endian byte order, the least significant byte is always the first
byte; the byte at the lowest address. With big-endian byte ordering, the
byte storage order in memory is dependent on the size of the data (byte,
word, longword) that is being referenced.

Endian-ness is a problem has been solved many times before. Some of the typical solutions include htonl/htons and ntohl/ntohs in the standard C library and the TCP/IP Services XDR (eXternal Data Representation) libraries. One of the more recently introduced network formats, and one that is seeing extensive press and marketing coverage, is XML.

10.22 How to resolve LINK-I-DATMISCH errors?

The message LINK-I-DATMISCH is informational, and indicates that the version of the specified shareable image found in the system shareable image directory does not match the version of the shareable image that was originally loaded into IMAGELIB.OLB, one of the OpenVMS libraries typically searched by the LINKER.

From a privileged username, you can usually completely repair this via the following DCL command:

```
$ LIB/REPLACE/SHARE SYS$LIBRARY:IMAGELIB.OLB SYS$SHARE:LIBRTL.EXE
```

This command assumes that the shareable image that was found in the SYS$SHARE: area is valid and upward-compatiable, and that the image has simply replaced an older version without also updating IMAGELIB.

10.23 VAX C and DEC C, and other OpenVMS C Programming Considerations?

VAX C V3.2 was released for OpenVMS VAX systems in 1991. DEC C V4.0 replaced VAX C V3.2 in 1993 as the HP C compiler for OpenVMS VAX systems. HP C is the ANSI C compiler for OpenVMS Alpha systems. VAX C predates the ANSI C standards, and has various areas that are not compliant with ANSI C requirements. HP C is an ANSI C compiler, and can also compile most VAX C code when /STANDARD=VAXC is specified. Versions of this compiler between V3.2 and V6.5 (exclusive) were known as DEC C, DIGITAL C, and Compaq C.

Both compilers can be installed at the same time on the same OpenVMS VAX system, allowing a migration from VAX C to DEC C, and allowing the same DEC C code to be used on OpenVMS VAX and OpenVMS Alpha.

The system manager can choose the system default C compiler when HP C is installed on a system with VAX C, and a C programmer can explicitly select the required compiler for a any particular compilation.

A current "C" license PAK allows access to both VAX C and HP C on the same OpenVMS VAX system.

Various HP C versions can be installed on OpenVMS VAX V5.5-2 and later. OpenVMS VAX releases such as V5.5-2 and V6.0 will require the installation of a HP C RTL kit, a kit that is included with the HP C compiler. OpenVMS VAX versions V6.1 and later do not require a seperate RTL kit, but HP C RTL ECO kits are available to resolve problems found with the C RTL on various OpenVMS releases.

With HP C, for automatic resolution of the standard C library routines by the LINKER utility, use the /PREFIX qualifier, such as /PREFIX=ALL_ ENTRIES. If a particular application program replaces an existing C library routine, use /PREFIX=(ALL_ENTRIES,EXCEPT=(...)). (VAX C required explicit specification of an RTL shareable image or C object library during the link.)

When the /PREFIX is requested, the compiler generates a "decc$" prefix on the specified symbols. This prefix allows the LINKER to resolve the external symbols against the symbols present in the DECC$SHR library. The DECC$SHR library is included in the IMAGELIB.OLB shareable image library, and IMAGELIB is searched by default when any program (written in any language) is LINKed. Because the standard C library routine names are very likely to match application routines written in other languages, a prefix "decc$" is added to the C symbol names to assure their uniqueness; to prevent symbol naming conflicts. C programs, however, can sometimes have private libraries for various purposes, and the external routines share the same names as the library routines. (This is not recommended, but there are applications around that use this technique.) Thus the need to explicity specify whether or not the "decc$" prefix should be prepended to the external symbol names by the compiler.

The qualifiers, and most (all?) with associated pragmas, that may be of interest when migrating VAX C code to HP C include:

- /PREFIX=ALL_ENTRIES

  As mentioned above. Failure to specificy this qualifier can cause the compiler to not add the prefixes for the names of the C library routines into the references placed in the object module, which can in turn cause problems resolving the external symbols in the library when the object code is linked.

- /ASSUME=WRITABLE_STRING_LITERALS

  Some VAX C programs erroneously write to the string literals. By default, HP C does not allow the constants to change.

- /SHARE_GLOBALS

  Enables sharing ("shr") of globals and of extern variables. HP C sets externs as non-shareable ("noshr"), VAX C as "shr".

- /EXTERN_MODE=COMMON_BLOCK

  VAX C assumes common block model for external linkages.

- /[NO]MEMBER_ALIGNMENT

  Refers to the padding placed between member elements within a struct. Disabling member alignment packs the data more tightly into memory, but this packaging has performance implications, both on OpenVMS VAX and particularly on OpenVMS Alpha systems.

Permit structure members to be naturally aligned whenever possible, and avoid using /NOMEMBER_ALIGNMENT. If you need to disable member alignment, use the equivilent #pragma to designate the specific structures. The alignment of structure members normally only comes into play with specific unaligned data structures—such as the sys$creprc quota itemlist—and with data structures that are using data that was organized by a system using byte or other non-member alignment.

Versions of HP C such as V6.0 include the capability to extract the contents of the standard header libraries into directories such as SYS$SYSROOT:[DECC$LIB...], and provide various logical names that can be defined to control library searches. With HP C versions such as V6.0, the default operations of the compiler match the expectations of most OpenVMS programmers, without requiring any definitions of site-specific library-related logical names. (And logical names left from older DEC C versions can sometimes cause the compiler troubles locating header files.)

HP C V5.6 and later include a backport library, a mechanism by which HP C running on older OpenVMS releases can gain access to newer RTL routines added to the RTL in later OpenVMS releases—the language RTLs ship with OpenVMS itself, and not with the compilers.

Example C code is available in SYS$EXAMPLES:, in DECW$EXAMPLES (when the DECwindows examples are installed), in TCPIP$SERVICES (or on older releases, UCX$EXAMPLES) when HP TCP/IP Services is installed), on the Freeware CD-ROMs, and at web sites such as

- http://www.openvms.compaq.com/wizard/

10–16

- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

10.23.0.0.0.1   Other common C issues

The localtime() function and various other functions maintain the number of years since 1900 in the "struct tm" structure member tm_year. This field will contain a value of 100 in the year 2000, 101 for 2001, etc., and the yearly incrementation of this field is expected to continue.

The C epoch typically uses a longword (known as time_t) to contain the number of seconds since midnight on 1-Jan-1970. At the current rate of consumption of seconds, this longword is expected to overflow (when interpreted as a signed longword) circa 03:14:07 on 19-Jan-2038 (GMT), as this time is circa 0x7FFFFFFF seconds since the C base date. (The most common solution is to ensure that time_t is an unsigned.)

If C does not correctly handle the display of the local system time, then check the UTC configuration on OpenVMS—the most common symptom of this is a skew of one hour (or whatever the local daylight savings time change might be). This skew can be caused by incorrect handling of the "is_dst" setting in the application program, or by an incorrect OpenVMS UTC configuration on the local system. (See section Section 4.4.)

When sharing variables with other languages, here is some example HP C code...

```
...
#pragma extern_model save
#pragma extern_model strict_refdef
extern int   VMS$GL_FLAVOR;
#pragma extern_model restore
...
```

and here is some associated example Bliss code...

```
...
EXTERNAL
   VMS$GL_FLAVOR,
....
```

10.23.0.0.0.2 Other common C++ issues

HP C++ (a separate compiler from HP C) provides both symbol mangling
and symbol decoration. Some of the details of working with longer symbol
names and the resulting symbol name mangling in mixed language
environments are listed in the shareable image cookbook, and in the
C++ documentation. Symbol name decoration permits the overloading of
functions (by adding characters to the external symbol for the function to
indicate the function return type and the argument data types involved),
and mixed-language external references can and often do need to disable
this decoration via the extern "C" declaration mechanism:

```
extern "C"
  {
  extern int ExternSymbol(void *);
  extern int OtherExternSymbol(void *);
  }
```

Also see Section 14.13 for information on /ARCHITECTURE and
/OPTIMIZE=TUNE.

See Section 10.16 for information on the C system and the lib$spawn call
in CAPTIVE environments.

Constructs such as the order of incrementation or decrementation
and the order of argument processing within an argument list are all
implementation-defined. This means that C coding constructs such as:

```
i = i++;
a[i] = i++;
foo( i, i++, --i);
```

are undefined and can have (adverse) implications when porting the
C code to another C compiler or to another platform. In general, any
combination of ++, −, =, +=, -=, *=, etc operators that will cause the
same value to be modified multiple times (between what the ANSI/ISO C
standard calls "sequence points") produce undefined and implementation-
specific results.

Within C, the following are the "sequence points": the ";" at the end of a C
statment, the | |, &&, ?:, and comma operators, and a call to a function.
Note specifically that = is NOT a sequence point, and that the individual
arguments contained within a function argument list can be processed
from right to left, from left to right, or at any random whim.

HP C for OpenVMS VAX (formerly DEC C) and VAX C do differ in the
related processing.

10–18

So you are looking for OpenVMS-specific definitions (include files)?

UCBDEF.H, PCBDEF.H and other OpenVMS-specific definitions—these are considered part of OpenVMS and not part of the C compiler kit—are available on all recent OpenVMS Alpha releases.

To reference the version-dependent symbol library sys$share:sys$lib_c.tlb, use a command similar to the following for compilation:

```
$ CC sourcea+SYS$LIBRARY:SYS$LIB_C/LIB
```

You can also define DECC$TEXT_LIBRARY to reference the library.

You will want to review the Programming Concepts manual, and specifically take a quick look at Chapter 21.

And some general background: the STARLET definitions (and thus the sys$starlet_c.tlb library) contain the symbols and the definitions that are independent of the OpenVMS version. The LIB definitions (and thus sys$lib_c) contain symbols and definitions that can be dependent on the OpenVMS version. You won't need to rebuild your code after an OpenVMS upgrade if you have included definitions from STARLET. The same cannot be said for some of the definitions in LIB—you might need to rebuild your code. (The UCB structure can and has changed from release to release, for instance.)

Recent versions of C automatically search sys$starlet_c.tlb. Explicit specification of sys$lib_c.tlb is required.

Also see the Ask The Wizard website topics (2486), (3803), and (1661):

- http://www.openvms.compaq.com/wizard/
- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

See Section 9.5 for information on the C off_t limitations, resolved in OpenVMS V7.3-1 and later and in ECO kits available for specific OpenVMS releases. The use of a longword for off_t restricts applications using native C I/O to file sizes of two gigabytes or less, or these applications must use native RMS or XQP calls for specific operations.

10.24   Status of Programming Tools on OpenVMS VAX?

DECthreads V7.3 and the HP C compiler (also known as Compaq C and DEC C) V6.4 are presently expected to be the last updates and the last releases of these development packages for use on OpenVMS VAX. The run-time support for both DECthreads (CMA$RTL) and for C (DECC$CRTL) will continue to be maintained, and will continue to be available on OpenVMS VAX. The VAX C V3.2 compiler is the final VAX C compiler release for OpenVMS VAX, and the VAX C Run-Time Library (VAXCRTL) will also continue to be available.

New development and new features and product enhancements continue for the OpenVMS Alpha and the OpenVMS IA-64 DECthreads and C compilers.

11 DECwindows

11.1 How do I let someone else display something on my workstation?

On a workstation, you will want to use the "Customize" menu of the session manager utility and select "Security". When the pop-up box appears, you can select the host node, username, and tranport that will allow you to launch an application that targets the workstation display.

If this does not provide you with access to the display, You need a checklist of sorts:

- Make sure that you've specified the X-windows "display" correctly on the remote host. For a DECnet transport, the specification uses two colons, while the TCP/IP transport typically uses one. The X Windows server and the X Windows screen follow the host specification, delimited by a period. For example:

Table 11–1 X Windows Display Commands

Shell	Command
csh	
	# setenv DISPLAY vms.domain:0.0
sh and ksh	
	# $ DISPLAY=vms.domain:0.0 ; export DISPLAY
DCL	
	$ SET DISPLAY/CREATE/NODE=vms.domain -/TRANSPORT=TCPIP/SERVER=server/SCREEN=screen

- If you have verified the command is correct and things are still not working, ensure the Security settings on the OpenVMS host side will allow the incoming connection: Pull down the "Options" menu in the Session Manager, and select "Security...". If you do not find your host

and username and transport listed among the authorized users, you will need to add an entry.

- There are various transports available, including LOCAL, DECNET, LAT, and TCPIP. You must Select the transport appropriate to the incoming connection.

- If the transport is "DECnet", do *NOT* add the double colon (::) to the node name.

- If the transport is "TCPIP", "Username" *must* be an asterisk (*). Why? Because unlike DECnet, the TCP/IP protocol does not provide the remote username information in the incoming connection.

- If the connection is "TCPIP", it is best to use a full domain name (e.g. Node.Subd.Domain). However, you may have to use the IP address itself, if your host does not have a way to resolve the address via DNS. If you have the luxury of fixed addresses (eg: you are *not* using DHCP), then it can be helpful to add two entries for each TCP/IP host, one that specifies the host name and one that specifies the host address.

- There are various TCP/IP packages for OpenVMS, and you must use syntax appropriate to the transport installed.

- If a TCP/IP connection is still not working, ensure that the transport you want has been activated for use with DECwindows. See Section 11.15 for details of configuring TCP/IP as a transport.

- There is a log file created in SYS$MANAGER: which can tell you which transports are loaded, and also tell you what connect attempts were rejected, including showing what the presented credentials were. This file is SYS$MANAGER:DECW$SERVER_0_ERROR.LOG, although the 0 could be another number if you have multiple servers on the workstation. I have found this file to be very useful for tracking down what needs to be put in the Session Manager Security entries.

11.2 How do I create a display on another workstation?

To create a display from an OpenVMS host to a remote X Windows display, use one of the following DCL commands:

```
$ SET DISPLAY /CREATE /TRANSPORT=net_transport /NODE=remote_node
$ SET DISPLAY /CREATE /TRANSPORT=LAT /NODE=remote_node
$ SET DISPLAY /CREATE /TRANSPORT=DECnet /NODE=remote_node
$ SET DISPLAY /CREATE /TRANSPORT=TCPIP /NODE=remote_node
```

Note that LAT is typically used only for the VXT series X Windows terminals, but it can also be used from OpenVMS to OpenVMS systems on various OpenVMS releases, such as on OpenVMS Alpha V6.1 and later. For details on configuring the TCP/IP transport, see Section 11.15.

If you are interested in X Windows terminals and have an older VAXstation system around, please see the EWS package on Freeware V5.0.

11.3 How can I get the information from SHOW DISPLAY into a symbol?

Use the undocumented SHOW DISPLAY/SYMBOL, and then reference the symbols DECW$DISPLAY_NODE, DECW$DISPLAY_SCREEN, DECW$DISPLAY_SERVER and/or DECW$DISPLAY_TRANSPORT.

An example of calling the underlying (and also undocumented) sys$qio programming interface for the WSDRIVER (WSAn:) is available at:

```
http://www.openvms.compaq.com/freeware/srh_examples/DECUS_UNDOC_CLINIC/
```

11.4 How do I get a log of a DECterm session?

If you are working from a Decterm, you can use the AutoPrint feature. Choose the "Printer..." menu item from the "Options" menu, set the printing destination to the name of the file you want, and set "Auto Print Mode". You are now free to continue.

It should be noted that all of the characters and escape sequences are captured, but if you display the resulting log file on a DECterm, then you will see exactly what was originally displayed.

11.5 Why does the DELETE key delete forward instead of backward?

This involves the Motif virtual key bindings. When a Motif application starts, it looks at the vendor string returned in the display connection information and attempts to match the string to a table of virtual bindings.

You can override the default bindings in your decw$xdefaults.dat file. Here is the entry you would make to get the default VMS bindings.

```
*defaultVirtualBindings:\
 osfCancel :   F11 \n\
 osfLeft :    Left \n\
 osfUp   :   Up  \n\
 osfRight :   Right \n\
 osfDown :   Down \n\
 osfEndLine :Alt  Right \n\
 osfBeginLine :Alt  Left \n\
 osfPageUp :   Prior \n\
 osfPageDown :   Next \n\
 osfDelete :Shift   Delete \n\
 osfUndo :Alt   Delete \n\
 osfBackSpace :   Delete \n\
 osfAddMode :Shift   F8  \n\
 osfHelp :   Help \n\
 osfMenu :   F4  \n\
 osfMenuBar :   F10 \n\
 osfSelect :   Select \n\
 osfActivate :   KP_Enter \n\
 osfCopy :Shift   DRemove \n\
 osfCut   :   DRemove \n\
 osfPaste :   Insert
```

To merge:

```
$ xrdb :== $decw$utils:xrdb.exe
$ xrdb -nocpp -merge decw$xdefaults.dat
```

Also note that the DECW$UTILS:DECW$DEFINE_UTILS.COM
procedure can be used to establish the xrdb and other symbols.

Also see the DECxterm directory of Freeware V5.0 for details on
connecting to OpenVMS from various UNIX platforms.

11.6 Why is DECwindows Motif not starting?

First check to see if there is a graphics device, usually a G* device. (eg:
On a DEC 2000 model 300, use the command SHOW DEVICE GQ) If you
do not find a graphics device:

*   OpenVMS has failed to find the appropriate IRQ information for
    an EISA graphics card (on the DEC 2000 series) such as the HP
    (Compaq) QVision, and did not autoconfigure it. Run the correct ECU
    (for Tru64 UNIX and OpenVMS) and reboot. This is necessary only
    on EISA-based systems.

- You have an EISA-based system (such as the DEC 2000 model 300) and do not have a HP (Compaq) QVision video card. This EISA graphics card should have Compaq printed on it, and identifies itself as a CPQ3011 or a CPQ3111. If it is not one of these two EISA devices, then OpenVMS does not support it. (There are no other supported EISA graphics controllers, and EISA graphics are normally used with DECwindows only on the DEC 2000 series systems.)

- You have a PCI-based system, and do not have a supported graphics controller—examples of supported controllers include the following:

  - PowerStorm 3D30

  - PowerStorm 4D20

  - PowerStorm 300

  - PowerStorm 350

  - ELSA GLoria Synergy

  - 3DLabs Oxygen VX1

  See Section 5.15 for further information on some of these graphics controllers.

- You have booted the system minimally, or have otherwise disabled the device autoconfiguration process.

If there is a G* graphics device present:

- There may have been a severe error in the DECwindows startup. Type the contents of SYS$MANAGER:DECW$SERVER_0_ERROR.LOG for any information on errors starting the server.

- The system parameter WINDOW_SYSTEM is not set to 1. While this was a common way for system managers to disable the DECwindows server startup, it is not particularly reliable as DECwindows can now "correct" this setting.

  If you really do not want an OpenVMS system with workstation hardware to bootstrap and configure itself as a workstation, add the following definition to SYLOGICALS.COM:

  ```
  $ DEFINE/SYSTEM/EXEC DECW$IGNORE_WORKSTATION TRUE
  ```

- You may not have a valid DECwindows Motif license loaded. To check for the two most common types of Motif product authorization keys (PAKs), use the following DCL commands:

```
$ LICENSE LIST DW-MOTIF/FULL
$ LICENSE LIST NET-APP-SUP*/FULL
```

and examine the information displayed. Make sure that one of these licenses is present, valid and active.

- Check that the DECW$PRIVATE_SERVER_SETUP.COM is correct for the graphics controller in use. For instance:

  The following is from the 9FX Vision 330 Owners Guide, EK-V330G-OG pg 2-9. Place the following in DECW$PRIVATE_SERVER_SETUP.COM, creatibng .COM from .TEMPLATE if necessary. Locate the DECW$PRIVATE_SERVER_SETUP.COM file in SYS$SPECIFIC:[SYSMGR] or in SYS$COMMON:[SYSMGR] as appropriate; the former file is used for this system within a cluster configuration, and the latter is used for all systems that do not also have a local copy of this file in SYS$SPECIFIC:[SYSMGR].

  ```
  $ DECW$XSIZE_IN_PIXELS == xvalue
  $ DECW$YSIZE_IN_PIXELS == yvalue
  $ DEFINE/SYSTEM DECW$SERVER_REFRESH_RATE rate_in_Hz
  ```

  Also see Section 11.12. Details of the PowerStorm 3D30 and 4D20 settings are available in the OpenVMS Ask The Wizard area.

11.7 How do I set the title on a DECterm window?

If you are creating a new DECterm window, check

```
$ HELP CREATE /TERMINAL /WINDOW_ATTRIBUTES
```

If you want to change the title of an existing window, use the following control sequences, where `esc` is the ANSI escape code, value decimal 27, and "text label" is what you want to display:

To set the DECterm title, send:

`esc`]21;text label `esc`\

To set the icon label, send:

`esc`]2L;text label `esc`\

To set both the DECterm title and icon to the full device name:

```
$  esc[0,7] = 27
$  fulldevnam = F$Edit(F$GetDVI("TT","FULLDEVNAM"),"UPCASE,COLLAPSE")
$  write sys$output esc+ "]21;" + fulldevnam + esc + "\"
$  write sys$output esc+ "]2L;" + fulldevnam + esc + "\"
```

You can also change the title and the icon using the Options-Window... menu.

Also see Section 12.1 and Section 8.11.

11.8 How do I customize DECwindows, including the login screen?

To customize various DECwindows Motif characteristics including the defaults used by the SET DISPLAY command, the DECwindows login screen background logo used (the default is the DIGITAL, Compaq, or HP logo), various keymaps, the FileView defaults, session manager defaults, the DECwindows login processing, DECwindows log file processing, and various other DECwindows attributes, see the example file:

```
$ SYS$STARTUP:DECW$PRIVATE_APPS_SETUP.TEMPLATE
```

This example template file is typically copied over to the filename SYS$COMMON:[SYS$STARTUP]DECW$PRIVATE_APPS_SETUP.COM and then modified to meet site-specific requirements.

Additionally, various X tools such as xsetroot, bitmap and xrdb—some these can be useful in customizing the appearance of an application or of the DECwindows Motif display—are provided in the DECW$UTILS: area.

When using DECwindows V1.2-4 and later on OpenVMS Alpha, the default desktop is the Common Desktop Environment (CDE). You can select your prefered desktop (CDE or DECwindows Motif) when logging in, or you can change the default to the DECwindows Motif desktop using the DCL symbol decw$start_new_desktop in the DECwindows private application setup command procedure. See SYS$STARTUP:DECW$PRIVATE_APPS_SETUP.TEMPLATE for further details, and how to create DECW$PRIVATE_APPS_SETUP.COM.

Note that with DECwindows CDE, the root window is no longer visible by default. The root window is hidden behind the "backdrop" window of the current CDE workspace. To make the root window visible, use the CDE style manager selection "backdrop none", and use information such as that in the OpenVMS FAQ to set the root window.

To add a new backdrop to the DECwindows CDE environment, the backdrop must first be in or be converted into X11 pixmap format. (This conversion is often possible using tools such as xv.) Then (if necessary) create the default backdrop directory SYS$COMMON:[CDE$DEFAULTS.USER.BACKDROPS]. Place the X11 pixmap file containing the desired image into the backdrops directory, ensure that it has a filename extension of .PM. (The xv default filename

extension for the X11 pixmap file is .XPM, while CDE expects only to see files with .PM.) Now invoke the CDE style manager and select a new backdrop. You will find your image will be placed at the end of the list of backdrops available.

11.9 Why doesn't XtAppAddInput( ) work on OpenVMS?

Yes, XtAppAddInput() does work on OpenVMS. The MIT definition of the X Windows call XtAppAddInput() includes platform-specific arguments.

On platforms where C is the typically the primary programming language for the platform, the file descriptor mask is one of the arguments to the XtAppAddInput() call.

On OpenVMS, the platform-specific arguments to this call include an event flag and an IOSB, as these are the traditional OpenVMS constructs used to synchronize the completion of asynchronous operations. While it would be easier to port non-OpenVMS C code that calls XtAppAddInput() over to OpenVMS if the arguments included the C file descriptor, this would make the call unusable from other OpenVMS languages, and would make it extremely difficult to use OpenVMS features such as ASTs and sys$qio calls.

One restriction on the event flag: the event flag chosen must be from event flag cluster zero. When using the traditional lib$get_ef and lib$free_ef calls to allocate and deallocate event flags, you must first explicitly call lib$free_ef to free up some event flags in event flag cluster zero. Please see the event flag documentation for specific details on these calls and for specific event flags that can be freed in event flag cluster zero.

Here is some example code that covers calling this routine on OpenVMS:

```
m->InputID = XtAppAddInput(
    m->AppCtx,
    m->InputEF,
    m->InputIosb,
    the_callback, 1 );
if ( !((int) m->InputID ))
    {
    XtAppErrorMsg(
        m->AppCtx,
        "invalidDevice",
        "XtAppAddInput",
        "XtToolkitError",
        "Can't Access Device",
        (String *) NULL,
        (Cardinal *) NULL );
    ...
```

11.10 Why do the keyboard arrow keys move the DECwindows cursor?

Congratulations, you have just stumbled into "dead rodent" mode. This DECwindows environment—where the keyboard arrow keys move the mouse cursor and where the [SELECT], [PREV], and [NEXT] keys emulate the three mouse buttons—allows rudimentary system operations when the mouse is among the casualties.

To enter or exit "dead rodent" mode, enter the following: `CTRL/SHIFT/F3`

11.11 Why does half my DECwindows display blank?

This is likely a result of receiving an OPCOM or other console message on a system that shares the system console with the DECwindows graphics workstation display.

You can toggle off the console display window using `CTRL/F2` and you can enable a serial console per Section 14.4 or Section 14.31.

Also see the console message window application available with recent DECwindows versions—DECwindows versions V1.2-3 and later will enable this window by default. For details on this console message window, see the DECW$CONSOLE_SELECTION option in SYS$STARTUP:DECW$PRIVATE_APPS_SETUP.TEMPLATE.

On older releases, you can disable output using the following:

```
$ SET TERMINAL/PERMANENT/NOBROADCAST OPA0:
$ DEFINE/USER SYS$COMMAND OPA0:
$ REPLY/DISABLE
```

Also see Section 14.30, Section 14.37, and Also see Section 8.3,

11.12 %DECW-W-NODEVICE, No graphics device found on this system?

To resolve the following error:

```
%DECW-W-NODEVICE, No graphics device found on this system
-DECW-I-NODECW, DECwindows graphics drivers will not be loaded
```

- Ensure that the system parameter WINDOW_SYSTEM is set to 1. If it is not set to a value of 1, issue the commands:

```
$ run sys$system:sysgen
USE CURRENT
SET WINDOW_SYSTEM 1
WRITE ACTIVE
WRITE CURRENT
EXIT
```

Then reboot the system.

- On OpenVMS Alpha, ensure the SYSMAN IO PREFIX LIST is set correctly, and specifically ensure the DECW$ prefix is included in the existing list. If it is not, you will need to add it:

```
$ run sys$system:sysman
IO SHOW PREFIX
IO SET PREFIX=(DECW$,*)   * = list returned by the show command
IO AUTO/LOG
EXIT
```

- Ensure that the image SYS$SHARE:DECW$ICBM.EXE is installed in memory. If it is not installed, then install it:

```
$ INSTALL LIST/FULL SYS$SHARE:DECW$ICBM
$ INSTALL REPLACE SYS$SHARE:DECW$ICBM
$ EDIT SYS$MANAGER:SYCONFIG.COM

$! The following line was added to install
$! support for the Mach64 Graphics Card
$!
$ INSTALL REPLACE SYS$SHARE:DECW$ICBM
$ ^Z
```

Then reboot the system.

The ICBM mechanism is not used on and not needed by more recent DECwindows versions.

- If the system still complains "%DECW-W-NODEVICE, No graphics device found on this system", then:

  - Boot the system as normal

  - Login as SYSTEM.

  - Create the file SYS$COMMON:[SYSMGR]DECW$USER_ AUTOCONFIG.DAT. Protection must permit world read access.

  - Add the following string on the very first line:

    ```
    CLEAR_PFLAG = ISA_4BYTE
    ```

  - Save the file

  - Set the file protections

    ```
    $ SET PROTECTION=W:RE SYS$MANAGER:DECW$USER_AUTOCONFIG.DAT
    ```

  - Reboot the system

  Also see Section 11.6.

11.13 How can I reset the warning bell volume?

With DECwindows CDE drivers and ECOs starting with ECOs for the DECwindows keyboard driver SYS$IKBDRIVER.EXE in OpenVMS Alpha V7.1-2 and V7.2-1 and with the SYS$IKBDRIVER.EXE included in OpenVMS V7.2-1H1 and later, the DECwindows CDE controls will now correctly manage the setting of the warning bell volume.

Unfortunately, the equivalent controls in the older DECwindows Motif interface are not compatible and can no longer manage the warning bell volume.

If you need to manage the volume with DECwindows Motif, consider using the following approach:

```
$ @decw$utils:decw$define_utils
$ xset b 1 100 100
```

The numerics are the volume, pitch, and duration, respectively.

Why? When OpenVMS first started supporting the PC-style keyboards, the X Windows Server and the keyboard driver interface did not support the pitch and duration, and neither did DECwindows Motif. The DECwindows keyboard driver was accordingly changed to use the volume from the keyclick setting (keyclick is not available in a PC-style keyboard) and the bell volume setting to control the pitch and duration.

DECwindows CDE does provide sliders for setting pitch and duration, so the keyboard driver and X Windows Server were modified to provide all of the information, and now the DECwindows CDE sliders work. This change is unfortunately incompatible with the old scheme used on the pre-CDE desktops, and the volume controls are now incompatible with the current keyboard drivers. Hence the use of xset.

11.14   How can I alter the DECwindows CDE backdrop?

To select a separate backdrop to be displayed on each screen using DECwindows CDE:

- Click on the Application Manager. This is the drawer icon on the CDE toolbar.

- Click on Desktop Tools

- Click on Set Default Screen and select the required screen

- Click on the Style Manager. This is the one containing the mouse and ttt on the CDE toolbar

- Now change the background.

11.15   How can I enable the DECwindows TCP/IP Transport

To configure the TCP/IP transport for DECwindows, first ensure that a TCP/IP package is installed and configured. Then set the DCL symbol DECW$SERVER_TRANSPORTS in SYS$MANAGER:DECW$PRIVATE_SERVER_SETUP.COM to the appropriate local value, based on the comments in that file. If you do not have a copy of SYS$STARTUP:DECW$PRIVATE_SERVER_SETUP.COM, the use the following COPY command to create this file based on the provided template file:

```
$ COPY SYS$MANAGER:DECW$PRIVATE_SERVER_SETUP.TEMPLATE -
$_ SYS$COMMON:[SYSMGR]DECW$PRIVATE_SERVER_SETUP.COM
```

12 Miscellaneous Information

12.1 Where can I find information on escape and control sequences?

Information on escape and control sequences can be found in the *OpenVMS I/O User's Reference Manual*, in the chapter on the terminal driver. The chapter also includes details on the general format and content of these sequences.

Specific details on the escape and control sequences supported by a particular serial device are typically found in the documentation provided with the specific device. Information on the sequences supported by DECwindows DECterm terminal emulator are included in the DECwindows documentation.

Examples of common escape and control sequences—those typically used by the OpenVMS screen management package—can be found in the OpenVMS system file SYS$SYSTEM:SMGTERMS.TXT.

The following refers to the function keys on the LK-series keyboards found on the VT-series terminals such as the VT220 and VT510, and the LK-series keyboards found on the OpenVMS workstations, and the keyboards found on compatible terminals. (Though note that the keyboard itself does *not* generate the sequence, the terminal or terminal emulator generates the sequence in response to user input.) In the following, {CSI} is decimal code 155 and can be replaced by the sequence "{ESC}[" (without the quotes) particularly for seven-bit operations, SS3 is decimal code 143 and can be replaced by "{ESC}O" particularly for seven-bit operations. Older VT1xx series terminals and any other terminals operating with seven-bit characters should not be sent eight-bit operators such as {CSI} and {SS3}.

```
PF1={SS3}P PF2={SS3}Q PF3={SS3}R PF4={SS3}S
KP0={SS3}p KP1={SS3}q KP2={SS3}r KP3={SS3}s KP4={SS3}t KP5={SS3}u
KP6={SS3}v KP7={SS3}w KP8={SS3}x KP9={SS3}y KPCOMMA={SS3}l KPMINUS={SS3}m
KPPERIOD={SS3}n ENTER={SS3}M DNARROW={CSI}B UPARROW={CSI}A LFARROW={CSI}D
RTARROW={CSI}C FIND={CSI}1~ INSERT={CSI}2~ REMOVE={CSI}3~ SELECT={CSI}4~
PREV={CSI}5~ NEXT={CSI}6~ F6={CSI}17~ F7={CSI}18~ F8={CSI}19~ F9={CSI}20~
F10={CSI}21~ F11={CSI}23~ F12={CSI}24~ F13={CSI}25~ F14={CSI}26~
HELP={CSI}28~ DO={CSI}29~ F17={CSI}31~ F18={CSI}32~ F19={CSI}33~ F20={CSI}34~
```

An example of working with escape sequences (in DCL) follows:

```
$ esc5m = "*[5m"
$ esc5m[0,8] = 27
$ esc0m = "*[0m"
$ esc0m[0,8] = 27
$ write sys$output esc5m + "blinking text" + esc0m
```

Documentation on an ANSI terminal relatively similar to the VT525 series is available at:

- ftp://ftp.boundless.com/pub/text/adds/docs/260_prog/

- ftp://ftp.boundless.com/pub/text/adds/docs/260_user/

Also see the various documentation and manuals available at:

- http://www.vt100.net/

Information on the ReGIS graphics character set is available at:

- http://www.cs.utk.edu/~shuford/terminal/dec_regis_news.txt

Also:

- http://www.boundless.com/Text_Terminals/VT/

Also see Section 11.7, Section 8.11.

12.2   Does DECprint (DCPS) work with the LRA0 parallel port?

The parallel printing port LRA0: found on many OpenVMS Alpha systems is capable of some bidirectional communications, with enough for basic operations with most parallel printers.

DECprint (DCPS) requires more than just the simple handshaking provided by the LRA0: port, therefore DCPS does not work with the LRA0: port.

12.3   How do I check for free space on a (BACKUP) tape?

You cannot know for certain, though you can certainly estimate the remaining capacity.

Tape media is different than disk media, as disks have a known and pre-determined fixed capacity. Modern disks also appear logically perfect, based on bad block revectoring support and the extra blocks hidden within the disk structure for these bad block replacements.

12–2

The capacity of tape media is not nearly as pre-determined, and the capacity can vary across different tape media (slightly different media lengths or different foil markers or other variations, for instance) and even on the same media over time (as bad spots in the media arise). Tapes can vary the amount of recording media required, depending on the remaining length of the tape, the numbers of correctable and uncorrectable media errors that might occur, the numbers and sizes of the inter-record gaps and related tape structure overhead, the particular media error recovery chosen, the tape density, the efficiently of any data compression in use, and the storage overhead required by BACKUP, tar, and other similar commands.

BACKUP using with the default settings results in approximately 15% overhead, in terms of saveset size. (eg: Assuming a 500 KB input, the total size would be 575 KB.)

Assuming no compression:
4 GB media / 575 KB saveset = 7294 savesets

Assuming 1:2 compression:
8 GB media / 575 KB saveset = 14588 savesets

Note: There are no inter-record gaps on DAT tapes. (When determining media capacity, you have to consider these with nine-track magtape media. Not with DAT (DDS). However, the block structure underneath the variable length record recording is based on a block size of circa 124 KB. Further, writing doubles filemarks and such can cause a loss of up to the underlying block size. Thus even though there are no inter-record gaps on DAT, larger savesets are still usually best.

The compression algorithms used on various devices are generally not documented—further, there is no way to calculate the effective data compression ratio, the tape mark overhead, and similar given just the data to be stored on tape—short of actually trying it, of course.

A typical compression ratio found with "everyday" data is somewhere around 1:1.8 to 1:2.

Note: OpenVMS often uses the term COMPACTION for compression control, as in the qualifier /MEDIA_FORMAT=COMPACTION.

12.4 So what happened to sys$cmsuper?

There is no SYS$CMSUPR service.

The typical wisdom for getting into supervisor access mode (from user mode) is to execute a routine in executive mode (via a call to SYS$CMEXEC and the appropriate privilege) and then issue a SYS$DCLAST with the ASTADR parameter pointing to your routine entry point and the ACMODE parameter specified as PSL$C_SUPER.

Alternatively, you can reset mode in the call stack return path and unwind from executive or kernel out into supervisor mode.

12.5 Correctly using license PAKs and LMF?

If you have multiple LMF$LICENSE.LDB databases in your OpenVMS Cluster, then each and every PAK must be installed in each and every license database present in an OpenVMS Cluster. Even if you use /EXCLUDE or /INCLUDE, you need to have a consistent set of PAKs registered across all licensing databases present in the OpenVMS Cluster.

If your software license permits it, you can use the following two commands to transfer license PAKs:

```
$ LICENSE COPY...
$ LICENSE ISSUE/PROCEDURE/OUTPUT=file product,...
```

To display the particular license(s) required (such as when you receive a NOLICENSE error), use the following DCL sequence:

```
$ SET PROCESS/PRIVILEGE=ALL
$ REPLY/ENABLE
$ DEFINE/SYSTEM/EXECUTIVE LMF$DISPLAY_OPCOM_MESSAGE
```

This logical name will cause all license failures to generate OPCOM messages, and this will hopefully show which license(s) you need— there may well also be additional license failures displayed, as various products can check for and can be enabled by multiple license PAKs. You will want to deassign this logical name when done.

Some of the more common license PAKs:

```
DECnet Phase IV:    DVNETRTG, DVNETEND, DVNETEXT, or NET-APP-SUP*
DECnet-Plus:        DVNETRTG, DVNETEND, DVNETEXT, or NET-APP-SUP*
TCP/IP Services:    UCX, or NET-APP-SUP*
OpenVMS Alpha:      OPENVMS-ALPHA and OPENVMS-ALPHA-USER
OpenVMS VAX:        VAX-VMS
OpenVMS Galaxy:     OPENVMS-GALAXY
Cluster (Alpha):    VMSCLUSTER, NET-APP-SUP*
Cluster (VAX):      VAXCLUSTER, NET-APP-SUP*
```

Various NET-APP-SUP (NAS) license packages are available, each with differing collections of products authorized. See the various NAS Software Product Description (SPD) documents for specific details.

- http://www.compaq.com/info/spd/
  OpenVMS typically uses SPD 25.01.xx and/or SPD 41.87.xx.

To determine which license PAK is failing (via a license check failure OPCOM message), use the command:

```
$ DEFINE/SYSTEM/EXECUTIVE LMF$DISPLAY_OPCOM_MESSAGE TRUE
```

Realize that defining this logical name will cause license checks that are otherwise hidden (unimplemented, latent, or part of a check for any of a series of licenses) to become visible. In other words, expect to see some spurious license check calls when you define this.

For information on PAKGEN and on generating license PAKs, please see Section 10.9.

13 Finding and using Software

13.1 Where can I find freeware/shareware/software for OpenVMS?

- Details on many commercial OpenVMS products are available in the catalog located at:

  - http://www.compaq.com/csa/directory/

- An OpenVMS Freeware CD-ROM is distributed with OpenVMS, and is also available separately (QA-6KZAA-H8). The contents of the OpenVMS Freeware CD-ROM media are also available online at:

  - http://www.openvms.compaq.com/freeware/

  - ftp://ftp.montagar.com/

  - ftp://mvb.saic.com/freewarev40/

  and at various other sites. The website also includes various updates and new packages that become available after the CD-ROM distributions are created.

  Submissions to the OpenVMS Freeware can be made via:

  - http://www.openvms.compaq.com/openvms/freeware/

  To acquire the OpenVMS Freeware CD-ROM distribution, you can order an OpenVMS distribution from HP (the Freeware is included)(see the OpenVMS SPD for part numbers), or you can specifically order a Freeware distribution from HP under part number:

  - QA-6KZAA-H8

  The Freeware CD-ROM set contains a large assortment of freeware, and is a good starting point if looking for utilities. Many of the packages listed below are also on the Freeware CD. Some of the most oft-requested OpenVMS tools on the Freeware CD include ZIP and UNZIP and GZIP (please see SOFT14), MMK (make), PINE, PERL, TAR, UUENCODE and UUDECODE. Many other tools are available on the Freeware.

- The UUENCODE and UUDECODE tools and various other tools are also available as part of TCP/IP Services package. (Use the DCL command procedure SYS$STARTUP:TCPIP$DEFINE_ COMMANDS.COM available on V5.0 and later to set up the necessary DCL foreign command symbols used for these and for various other tools provided by TCP/IP Services.)

- OpenVMS software (formerly at Western Kentucky University (WKU) is now available via Madgoat and via Process Software archives:

  - http://www.process.com/openvms/index.html

  - http://www.madgoat.com/

- The FILESERV packages are also available via anonymous FTP from:

  - ftp://ftp.process.com/vms-freeware/

  - ftp.process.com, under [.WKU.VMS.FILESERV].

  - ftp.vms.stacken.kth.se, under [.MIRRORS.WKU.VMS.FILESERV].

  - ftp.ctrl-c.liu.se, under [.WKU.VMS.FILESERV].

  - ftp.riken.go.jp

  - ftp.vsm.com.au, under kits and kits/decwindows.

  - ftp.vsm.com.au, via the WWW instead of FTP.

  The packages are also available via e-mail from FILESERV@PROCESS.COM. Send the commands HELP and DIR ALL in the body of a mail message to this email address for further information on retrieving the files.

- Another source of free software is the vmsnet.sources newsgroup (and the corresponding vmsnet.sources.d discussion group). See the monthly posting "vmsnet.sources archives" for a list of sites which archive submissions to vmsnet.sources.

- Arne Vajhøj runs an OpenVMS WWW page, with software and other pointers, at:

  - http://www.levitte.org/~ava/

- The Kermit file transfer package is available at:

  - http://www.columbia.edu/kermit/

  - ftp://kermit.columbia.edu/

- ZMODEM is available at:

  - ftp://ftp.cs.pdx.edu/pub/zmodem

  See the FILES file in that directory for further details. Note that this freeware version of ZMODEM will interoperate only with ZMODEM software that is licensed from Omen Technology. (This package is also available on various of the Freeware distributions.)

- A good source of software for OpenVMS systems and for various other platforms is the Encompass (formerly known as DECUS) library. Online catalogs are available as well as some software via

  - ftp://ftp.decus.org

  - gopher://gopher.decus.org/

  - http://www.decus.org/

- DECUS SIG Tape collections are available on Mark Berryman's system,

  - ftp://mvb.saic.com/

- David Jones's DECthreads-based HTTP_SERVER World-Wide Web server for OpenVMS, often known as the OSU webserver:

  - http://kcgl1.eng.ohio-state.edu/www/doc/serverinfo.html

- Secure Shell (SSH) Server for OpenVMS:

  - http://kcgl1.eng.ohio-state.edu/~JONESD/ssh/DOC/

  Secure Shell (SSH) Client for OpenVMS:

  - http://www.free.lp.se/fish/

  Information on OpenSSL (SSLeay) for OpenVMS:

  - http://www.free.lp.se/openssl/

  The SSH terminal client FISH:

  - http://www.free.lp.se/fish/

  Information on OpenSSL (SSLeay) and OSU Web server interoperation:

  - http://www.ourservers.net/openvms_ports/

  OpenSSL is included with OpenVMS V7.3-1 and later.

- DECwindows Motif V1.2-3 includes NCSA Mosaic 2.4 built for UCX. V1.2-4 includes Spyglass Enhanced Mosaic, which supports many "Netscape" enhancements. Netscape Navigator and Mozilla are also available for OpenVMS.

- A port of Mosaic supporting UCX, Multinet and SOCKETSHR/NETLIB is available from:

  - ftp://wvnvms.wvnet.edu/mosaic/

  Versions of Mosaic are also available on the Freeware.

- Lynx (a character-cell World-Wide-Web reader) is available from

  - ftp://ftp2.cc.ukans.edu/pub/lynx

  Versions of Lynx are also available on the Freeware.

- The Netscape Navigator and Mozilla web browsers are available at:

  - http://www.openvms.compaq.com/ebusiness/Technology.html

  - http://www.mozilla.org/

- PGP (Phil Zimmerman's "Pretty Good Privacy") is available from various distribution sites, including those listed in the PGP FAQ. Information on PGP and on OpenVMS downloads of PGP is available. Relevent URLs include:

  - http://www.ipgpp.org/

  - http://web.mit.edu/network/pgp.html

  - http://www.pgpi.com/

  - http://www.yrl.co.uk/~phil/pds/pds.html

  - http://www.nai.com/

- GNU Privacy Guard (GPG, GnuPG) is available.
  Search the comp.os.vms newsgroup archives for information regarding GnuPG; the source code, binaries for various platforms, and documentation are all available at:

  - http://www.gnupg.org/

  The OpenVMS source code and OpenVMS Alpha images can be found at:

  - http://saf.bio.caltech.edu/pub/SOFTWARE/openvms/

  As of this time, the port only runs on OpenVMS Alpha, but an investigation of an OpenVMS VAX port is reportedly under consideration.

- An archive of the CENA DECwindows, X Windows, and VMS software packages can be found at the following sites:

  - http://decwarch.free.fr/

13–4

- ftp://ftp2.cnam.fr/decwindows/
- ftp://ftp.ctrl-c.liu.se/decwindows/
- ftp://ftp.vms.stacken.kth.se/mirrors/decwindows/
- http://www.multimania.com/pmoreau/decw/

Other FTP mirror sites for the DECwindows archive include:

- axp.psl.ku.dk (Multinet)
- ftp2.cnam.fr (MadGoat)

ftp.x.org (in /contrib/vms) is effectively not a mirror, but it does have various OpenVMS packages from the DECwindows archive.
A list of the http mirror sites for the DECwindows archive is available at:

- http://axp616.gsi.de:8080/wwwar/cena/decwindows/cena.html

Various of these packages are also available on the Freeware.

- ImageMagick is an X11 package for display and interactive manipulation of images. The package includes tools for image conversion, annotation, compositing, animation, and creating montages. ImageMagick can read and write many of the more popular image formats (e.g. JPEG, TIFF, PNM, XPM, Photo CD, etc.).

  - http://www.imagemagick.org/

  Versions of ImageMagick are also included on the Freeware.

- XV is a image viewing and editing tool and is available from:

  - ftp://ftp.cis.upenn.edu/pub/xv

  - http://www.sanface.com/

  - ftp://www-pi.physics.uiowa.edu/~dyson/xv-3_10a-vms.zip

- Many software packages are available for displaying various bitmap files (.JPG, .GIF, .BMP, etc) on OpenVMS. Xloadimage, Xli, Xv, ImageMagick are the most common tools used under OpenVMS. Various web browsers such as Mozilla (qv) can also display various file formats on OpenVMS. You can find some of these tools at the DECwindows Archive:

  - http://www.multimania.com/pmoreau/decw/index.html

  - http://www.multimania.com/pmoreau/decw/images.html

- GHOSTSCRIPT (gs) and GHOSTVIEW (gv) are available from:

  - http://www.openvms.compaq.com/freeware/freeware50/

13–5

Versions are also available on the Freeware.
Also see:

- http://www.cs.wisc.edu/~ghost/gnu/

- XPDF, a viewer for PDF (Adobe Acrobat) files, is available from:

  - http://www.foolabs.com/xpdf/

  Also available on the Freeware.

  Beware: the XPDF tool included on OpenVMS Freeware V4.0 is dated, and has been found to have various bugs. Use the Freeware V5.0 version of the XPDF kit, or later.

  A Java-based PDF viewer is available from Adobe, and is known to operate on recent OpenVMS Alpha releases:

  - http://www.adobe.com/

- Various OpenVMS-related tools—both freeware and shareware—such as txt2pdf—are available from at:

  - http://www.sanface.com/

- The MPEG library version 1.1 is available for OpenVMS VAX and Alpha at

  - ftp://ftp.x.org/contrib/vms/mpeglib-11-vms.readme

  - ftp://ftp.x.org/contrib/vms/mpeglib-11-vms.zip

  Various MPEG players and related tools are available on the Freeware.

- An Archie clone is available at INFO.CS.PUB.RO. Telnet to that machine, and login as VMSARCI. It contains now listings for over 30 ftp servers with over 14 gigabytes of OpenVMS software.

  The most useful commands are LIST, which generates a list of scanned ftp servers, and FIND string, whichs looks for a file containing "string" in the name; the search modes are only "substring" [default] and "exact", and regex search is not supported (so FIND EMACS will work, but FIND *EMACS* or FIND *EMACS*.* will not). The search is case-insensitive.

  The maintainer of the site (stfp@roipb.cs.ipb.ro) requests that anyone submit notifications of FTP servers with OpenvMS software that are not listed on the site.

- The Levitte (extended :-) Family (and OpenVMS) website:

  - http://www.levitte.org/

- http://www.levitte.org/~ava/
- Robert Alan Byer's OpenVMS Hobbyist Systems, including OpenVMS public domain software and various ports:
  - http://www.ourservers.net/
- CalTech Software Archives:
  - http://seqaxp.bio.caltech.edu/pub/SOFTWARE/AAA_ CONTENTS.TXT
- DJE Systems Website (David J. Dachtera)
  - http://www.djesys.com/freeware/vms/
- Web Servers:
  Apache Web server (HP (Compaq) Secure Web Server (CSWS)):
  - http://www.openvms.compaq.com/ebusiness/Technology.html
  - http://www.er6.eng.ohio-state.edu/~jonesd/apache/1_3_9/
  OSU Web server
  - http://www.er6.eng.ohio-state.edu/www/doc/serverinfo.html
  - http://www.kjsl.com/archives/
  - email list: VMS-WEB-daemon-Request@KJSL.COM
  WASD Web server
  - http://wasd.vsm.com.au/wasd/
  Purveyor Web server:
  - email list: listserv@cjis.ci.lincoln.ne.us, no subject, one line message: SUBSCRIBE PURVEYOR
- CD-R (CD-Recordable) media tools are available, please see Section 9.7.
- Grace (WYSIWYG 2D plotting tool)
  - http://plasma-gate.weizmann.ac.il/Grace/
- The POV-Ray ("Persistance of Vision" Raytracer) ray-tracing graphics package is available on the OpenVMS Freeware.
- Majordomo mailing list handler:
  - http://www.openvms.compaq.com/ebusiness/Technology.html

- PINE (OpenVMS tools for sending and receiving MIME mail):
    - ftp://ftp2.kcl.ac.uk/pub/vms/pine-vms/
    - http://www.agh.cc.kcl.ac.uk/files/vms/pine-vms/
    A MIME tool is available in OpenVMS V7.2 and later.
    Also see the mmencode base64 encode and decode available at:
    - http://nucwww.chem.sunysb.edu/htbin/software_list.cgi
- Menufinder (menu-driven system management environment):
    - http://www.itre.com/mf/download.html
- tcgmsg, pvm, mpi, linda:
    - ftp://v36.chemie.uni-konstanz.de/tcgmsg_vms/tcgmsg_vms.zip
- OpenVMS software that can control a Tripp-Lite Uninterruptable Power Supply (UPS) is available from:
    - http://seqaxp.bio.caltech.edu/pub/SOFTWARE/TCONTROL.ZIP
    UPShot web-based software for controlling a UPS is available from:
    - http://www.tmesis.com/apc/beta.htmlx
- Examples of using the OpenVMS Foreign MAIL interface are available at:
    - http://www.hhs.dk/anonymous/pub/vms/collection/foreignmail.zip
    - http://www.hhs.dk/anonymous/pub/vms/nbl/nbl.zip
- For tools to manage or to search your OpenVMS MAIL file, see:
    - http://vms.process.com/scripts/fileserv/fileserv.com?MLSEARCH
- AscToHTM attempts to convert any plain text file to HTML, while AscToTab restricts itself to files that are plain text tables. (Versions are also availabe on the OpenVMS Freeware).
    - http://www.jafsoft.com/asctohtm/index.html
    - http://www.jafsoft.com/asctotab/index.html
- Information on the SAMBA package, a package that provides SMB-based Microsoft Windows PC disk and print services for OpenVMS, is available at:
    - http://ifn03.ifn.ing.tu-bs.de/ifn/sonst/samba-vms.html
    - http://www.samba.org/

To subscribe to the SAMBA-VMS mailing list e-mail listproc@samba.org with no subject line and the following single line of text:

```
subscribe samba-vms Your Full Name
```

Also see:

- http://lists.samba.org/

- The Perl language is available for OpenVMS, see Section 13.10 for details.

- XML is available for OpenVMS.
  Source code of an XML Parser is available from Oracle.
  Also see:

  - http://www.python.org/sigs/xml-sig/

  An XML parser is available as part of OpenVMS V7.3 and later.

- GTK+ (The GIMP GUI Tookit) for OpenVMS:

  - http://www.openvms.compaq.com/ebusiness/Technology.html

- The OpenVMS Porting Library now available, and is intended to permit easier porting of C and C++ applications from UNIX systems to OpenVMS:

  - http://www.openvms.compaq.com/ebusiness/Technology.html

  GTK is also available.

- Mlucas (specialized FFT):

  - ftp://hogranch.com/pub/mayer/README.html

- Tools to monitor the terminals and the activity of other OpenVMS users (in addition to existing auditing capabilities in OpenVMS) are available. Peek and Spy (Networking Dynamics) and Contrl (Raxco) are two of the commercial packages, while the freeware Supervisor package is available on OpenVMS VAX.

  - http://www.networkingdynamics.com/

  - http://www.raxco.com/

- Python for OpenVMS:

  - http://www.python.org/

  Also see the OpenVMS Freeware.

- Various packages for OpenVMS:
  - http://richj.home.mindspring.com/richware/index.html
  - http://www3.sympatico.ca/n.rieck/links/cool_vax_vms.html
- TSM (Terminal Server Manager) is available via:
  - http://www.compaq.com/support/digital_networks_
    archive/servers/tsm/index.html
  - http://www.openvms.compaq.com/freeware/freeware50/
  - Beware: The TSM saveset shipped on the Freeware V5.0 disk
    media is busted. Download a new copy of the saveset from the
    Freeware website.
- TCL for OpenVMS:
  - ftp://sapodilla.rsmas.miami.edu/pub/VMS-tcl/tcl8.0p2-tk8.0p2/
- make, gmake, mmk and other build tools are available on the
  Freeware.
- An OpenVMS port of the ht://Dig web search engine is available at:
  - ftp://ftp.pdv-systeme.de/vms/
- If you need to change the file modification date and are looking for a
  utility such as the UNIX touch tool, look at DFU on the OpenVMS
  Freeware (DFU SET or simular), or use an existing DCL commands
  such as:

  ```
  $ SET FILE/PROTECT=(current_protection_mask) [...]*.*
  ```

- A table listing translations between UNIX shell and OpenVMS DCL
  commands was posted to comp.os.vms by Christopher Smith some
  time ago. This page should be available from the google newsgroup
  archives.
- The UNIX touch tool is available via various means:

  ```
  $ RENAME filename.ext;version *
  ```

  - http://nucwww.chem.sunysb.edu/helplib/@hvmsapps/TOUCH
  MadGoat FILE tool (see the MadGoat archives)

  - use /REVISION_DATE or /CREATION_DATE
  The DFU tool (see the OpenVMS Freeware)

The pair:

```
$ set file 'p1' /acl=(ident=[system],access=none)
$ set file 'p1' /acl=(ident=[system],access=none) /delete

$ SET FILE/VERSION=0
```

The following hack:

```
$!    Command procedure SETDATE.COM
$!
$!    Changes the DATES for an input file to a
$!    file named OUTFILE.
$!
$assign/nolog 'p1' outfile
$convert/fdl=sys$input 'p1' outfile:
date
creation   01-apr-2010
expiration 01-Apr-2012
revision   01-Apr-2011
backup
...
```

The following RMS system service sequence:

```
$open (with xabrdt)
set the desired values in the xabrdt
$close
```

13.2   Where can I find UNIX tools for OpenVMS?

13.2.0.0.0.1   POSIX-compliant versions of POSIX routines and utilities

lex, yacc, grep, tar, uuencode, uudecode, rcs, man, cpio, make, awk, ar, mail, etc., the POSIX shell, the POSIX C programming interface, etc.

POSIX utilities can be used from within the POSIX shell, and via the DCL 'POSIX/RUN POSIX$BIN:tool.' command.

POSIX is a separately-installed package, and is licensed with OpenVMS V5.5 later. The POSIX installation kit is included on the consolidated distribution CD-ROM kit, and installation kits are also available separately.

The POSIX package is no longer supported on OpenVMS, components of the POSIX standard such as parts of the POSIX API are being added into OpenVMS. Versions of POSIX generally do not operate on V7.x OpenVMS VAX and OpenVMS Alpha releases.

13.2.0.0.0.2   C system and library routines

Common C system and library routines are present in the DEC C run-time library, which is available for V5.5 and later, and is shipped in V6.1 and later. DEC C is the upgrade for VAX C, DEC C and VAX C can coexist on the same system OpenVMS VAX system, and both compilers can be enabled via the "C" license PAK.

Also see SYS$EXAMPLES:, and (if either is installed) the DECW$EXAMPLES: and TCPIP$EXAMPLES (or prior to V5.0, UCX$EXAMPLES:) areas.

13.2.0.0.0.3   X Windows utilities and routines

Various X Windows utilities are available for DECwindows. DECwindows is an implementation of the X Windows environment and libraries, and provides various libraries, and provides various desktop interfaces, including COE, Motif, and XUI.

xwd, xev, mosaic WWW browser, xrdb, bmtoa and atobm, xpr, ico, etc. are available. Look in DECW$UTILS: in DECwindows Motif V1.2-3 and later. Also see DECW$EXAMPLES: for example X and C programs.

Miscellaneous tools and examples are also available. Examples include the older DWAUTH (X Windows SYSUAF authorize-like tool) tool, various versions of grep, fgrep, yacc, vmstar, uuencode, gawk, etc. html tools, the mx SMTP mail exchange package, X windows flight simulator, the mxrn X windows news reader, the OSU HTTPD WWW server, a WWW gopher browser, etc. are all on the various Freeware distributions.

13.2.0.0.0.4   TCP/IP Tools and Utilities for OpenVMS?

TCP/IP Services (formerly known as UCX) contains tools such as ping, uuencode, smtp, snmp, rcp, nfs, tnfs, etc.

OpenVMS V6.2 and later includes DCL-integrated support for various IP tools, with commands such as SET HOST/TELNET, and COPY/FTP. This interface requires the installation of an IP stack, and UCX V3.3 and later as well as any current third-party IP stack can be used. Once the IP stack is installed and configured, the DCL command qualifiers such as /FTP, /RCP, /RLOGIN, /TELNET, and /TN3270 are available on the various DCL commands.

Various C programming examples in TCPIP$EXAMPLES and (on releases prior to V5.0) in UCX$EXAMPLES:.

13–12

13.2.0.0.0.5   The vi text editor

vile, vim and elvis are all clones of the vi text editor, and all operate on OpenVMS.

Versions of vile are available on the Freeware and at:

• http://www.clark.net/pub/dickey/vile/vile.html

vim: vi improved

• http://www.polarfox.com/vim/

13.2.0.0.0.6   Various GNU tools

Information on the GNU on VMS (GNV) Project, which aims to port GNU software (BASH, etc) to OpenVMS, is available at:

• http://gnv.sourceforge.net/

Software info:

• http://vms.gnu.ai.mit.edu/software/

Software archive:

• ftp://vms.gnu.ai.mit.edu/gnu-vms/software/

Various GNU tools are also available on the Freeware.

13.2.0.0.0.6.1   GCC compiler

A mirror for work performed at the Progis company in Germany in porting GCC (GNU C) to OpenVMS Alpha and OpenVMS VAX is available at:

• ftp://vms.gnu.org/progis_mirror/gcc

• http://www.levitte.org/~ava/vms_gnu.htmlx

• ftp://ftp.caltech.edu/pub/rankin/

There are also updated header files for GCC on OpenVMS VAX that allow it to work with TCP/IP Sockets and the HP C RTL at:

• ftp://ftp.qsl.net/pub/wb8tyw/gcc281_u/

The HP C compiler and other development tools are part of the OpenVMS Hobbyist licensing program for non-commercial users, and these and other tools are available to commercial developers via the CSA partner program.

13.2.0.0.0.7   Console Management Options

Some of the available console management options for OpenVMS:

- http://www.robomon.com/ (Heroix)

- http://www.ki.com/products/clim/ (KI Products)

- http://www.globalmt.com/ (Global Maintech)

- http://www.tditx.com/ (TECsys)

- http://www.cai.com/products/commandit.htm (CA)

13.3   Where can I get the Netscape Navigator Mozilla.org Web Browser?

Mozilla.org is the open source organization providing Netscape and other interested parties with a browser. Netscape is expected to commercialize this mozilla.org browser, add additional proprietary features, and release it as Netscape Communicator (version number TBD). This Netscape Communicator will contain the features that the Internet community expected to see in Netscape Communicator V5.

OpenVMS Engineering is continuously porting Mozilla.org's web browser to OpenVMS—OpenVMS baselevels of the browser are currently available for download.

The Mozilla.org web browser schedule is available at:

- http://www.mozilla.org/

The latest information and current downloads are available at:

- http://www.openvms.compaq.com/ebusiness/Technology.html

See section Section 13.8 for information on various certificates for V3.003 Netscape Navigator; certificates that have presently expired.

13.4 Where can I get Java for OpenVMS?

Java is available on and is included with OpenVMS Alpha, starting with the OpenVMS Alpha V7.2 and later releases. Java download kits are available for OpenVMS Alpha V7.1 and later releases.

Java is not available on OpenVMS VAX. As for why: the Java language definition requires a floating point format (IEEE) that is not native to VAX, and this would require the emulation of all floating point operations within Java applications. Further, the C source code used to implement for Java itself is heavily dependent on passing IEEE floating point values around among the many internal subroutines, and adding support for VAX would entail changes to the HP C compiler for OpenVMS VAX— and specifically to the VAX VCG code generator that is used by HP C on OpenVMS VAX systems—in order to add support for passing IEEE-format floating point doubles around. Alternatively, extensive changes to the Java source code to remove the assumption that the double is an IEEE floating point value.

There are currently no plans to make a version of Java available for OpenVMS VAX. (A prototype version of Java was created for OpenVMS VAX, and performance was found to be inadequate. At best.)

If Java2 or other environment lifts the requirements for IEEE floating point as part of the language definition, this decision may be revisited.

If you are having problems with Display Postscript, you need to upgrade your Java kit—1.2.2-3 and later remove the requirement for Display Postscript extensions, and Java 1.2.2-3 is required with DECwindows V1.2-6 and later.

For additional information on Java for Alpha systems, please see the OpenVMS documentation (V7.2 and later), and the following site:

- http://www.compaq.com/java/alpha/index.html

HP (Compaq) Secure Web Server (CSWS) includes CSWS_JAVA, which provides the following Apache Tomcat technologies: JavaServer Pages 1.1, Java Servlet 2.2, and MOD_JK. (CSWS is based on the Apache web server. See SOFT1.)

13.5 Obtaining user input in DCL CGI script?

If you choose to use the GET method, then the form data is available in the DCL symbol QUERY_STRING, in URL-encoded format.

If you use the POST method, then you need to read the form data from stdin. For a DCL CGI script running under the Netscape FastTrack web server, you can read the data using the following READ command:

```
$ READ SYS$COMMAND postdata
```

to read the information in.

The following describes the use of DCL command procedures as CGI scripts with the OSU web server:

• http://www.levitte.org/~ava/cgiscripts_other.htmlx

DCL CGI is also discussed in the Writing Real Programs in DCL book, and in the Ask The Wizard website.

13.6 How can a batch job get its own batch entry number?

To have a batch procedure retrieve its own batch entry number, use the following:

```
$ Entry = F$GETQUI("DISPLAY_ENTRY", -
    "entry_number","display_entry","this_job")
```

Remember that the entry numbers issued by the OpenVMS Job Controller are opaque longword values. Don't assume you know the format of the number, nor the range of numbers you might see...

13.7 How do I convert to new CMS libraries?

A change was made to the format of the CMS database for CMS libraries starting with V3.5-03—to ensure that earlier versions of CMS are unable to access the database once the "conversion" to V3.5-05 and later is made, you must issue the following two commands when upgrading from V3.5-03 and prior. (The only differences between CMS version V3.5-03 and CMS version V3.5-05 involve changes to ensure that np earlier version of CMS can access the "converted" database.)

To perform the "conversion", issue the following commands for each CMS library present:

```
$ RENAME disk:[directory]00CMS.* 01CMS.*
$ COPY NLA0: disk:[directory]00CMS.CMS
```

13–16

The new file 00CMS.CMS must have the same security settings as the 01CMS.CMS file, and is created solely to ensure continued compatibility with tools that expect to find a 00CMS.CMS file (eg: various versions of the Language-Sensitive text editor LSEDIT).

13.8   Where can I get new certificates for Netscape Navigator?

The URLs that I found for adding/updating root certificates are:

- http://www.entrust.net/customer/generalinfo/import.htm entrust

- http://www.thawte.com/ thwate

- https://www.verisign.com/server/prg/browser/root.html verisign

To update certificates in Netscape Navigator V3.03 on OpenVMS, use the following:

Thawte Server certificate which expired in 1998:

1   Under the Options Menu choose "Security Preferences..."

2   Select the "Site Certificates" tab

3   Select "Thawte Server CA" in the list of certificates

4   Select "Delete Certificate" and then "OK"

5   Go to http://www.thawte.com/serverbasic.crt

6   Follow the instructions on the popup dialog box to accept the certificate This mostly involves hitting the "Next" button and clicking an accept button and then naming the resulting certificate. I named it the same name as the original.

VeriSign/RSA Server certificate which expired Dec 31, 1999:

1   Under the Options Menu choose "Security Preferences..."

2   Select the "Site Certificates" tab

3   Select "Verisign/RSA Secure Server CA" in the list of certificates

4   Select "Delete Certificate" and then "OK"

5   Go to https://www.verisign.com/server/prg/browser/root.html

6   Follow the instructions on the popup dialog box to accept the certificate This mostly involves hitting the "Next" button and clicking an accept button and then naming the resulting certificate. Verisign suggests using the name "VeriSign CA".

13.9 Why doesn't DCL symbol substitution work?

The DCL symbol substitution processing occurs only at the DCL prompt, not within data and not within files. If you wish to perform symbol substitution in this environment, you typically write a small file containing the command(s) and data to be invoked—potentially only the data—and you then invoke the created procedure or reference the specified data.

In this case, use of a file containing nolinemode commands or other techniques might be useful—you will want to ensure that the text editor you use does not attempt to use screen mode or similar, as this is not generally considered adventageous within a command procedure.

Tools such as FTP have alternatives: COPY/FTP.

DCL symbol substitution occurs in two passes, using the ampersand and the apostrophe. In most cases, only the apostrophe is necessary. In a few cases—such as the DCL PIPE command—you will may need to use the ampersand to get the substitution to work. The following example uses ampersand substitution to transfer the contents of the header into a logical name:

```
$ PIPE CC/VERSION | (READ SYS$PIPE hdr ; DEFINE/JOB/NOLOG hdr &hdr )
```

A logical name (in the job logical name table; shared by all processes in the current job) was used as DCL symbols cannot be returned back out from a DCL PIPE or other spawned subprocess.

13.10 Where can I get Perl for OpenVMS?

OpenVMS support is included in the standard distribution of Perl, the popular scripting language created by Larry Wall. In addition to nearly all of the functionality available under Unix, OpenVMS-specific Perl modules provide interfaces to many native features, as well as access to Oracle, Ingres, and Sybase databases via the Perl DBI available on OpenVMS.

A website useful for getting started with Perl on OpenVMS—where you will find such things as download links, instructions, auxiliary tools, and sample scripts—is available at:

• http://www.sidhe.org/vmsperl

If you have a C compiler, the best way to obtain Perl is to download and build it yourself. The latest production quality source kit is available from:

- http://www.perl.com/CPAN/src/stable.tar.gz

You will need GUNZIP and VMSTAR (both available from the OpenVMS Freeware CD, or from other sites) to unpack the archive; once you've done that, read the instructions in the README.vms file.

Binary distributions for most Alpha and VAX environments are available on the OpenVMS Freeware CD-ROM and from various websites, including the following:

- http://www.sidhe.org/vmsperl/prebuilt.html

- http://www.openvms.compaq.com/freeware/perl5/

During active Perl development cycles, test kits are sometimes found at: from:

- ftp://ftp.sidhe.org/

Watch the mailing list (see below) for details on experimental releases.

Charles Lane maintains pages on how to write CGI scripts in Perl for the OSU HTTP server, as well as more general tips, tricks, and patches for building and running Perl on OpenVMS:

- http://www.crinoid.com/crinoid.htmlx

There are OpenVMS-specific Perl modules that implement interfaces to a subset of the VMS System Services. With these modules, you can get (and often set) device, job, queue, user, system, and performance information. The lock manager, RMS indexed files, screen management utilities, and Intracluster Communication Services are also accessible via Perl. The relevant modules are all available from:

- http://www.perl.com/CPAN/modules/by-module/VMS

To subscribe to the OpenVMS Perl mailing list (a discussion forum for both user support and new development), send an email message to vmsperl-subscribe@perl.org

The mailing list archives may be searched at:

- http://www.xray.mpe.mpg.de/mailing-lists/vmsperl

13.11 Where can I get DECmigrate (VEST and TIE)?

The DECmigrate image translation tool, a tool that translates OpenVMS VAX images for use on OpenVMS Alpha is available at:

• http://www.support.compaq.com/amt/decmigrate/index.html

This tool is not currently supported.

See Section 7.4.

13.12 Where can I get Zip, Unzip, self-extracting zip, etc?

Many packages are provided in ZIP, GZIP, or BZIP2 format, which requires you to acquire the associated unzip tool to unpack it. You can get ZIP and UNZIP and related and similar tools from the following areas:

• http://www.openvms.compaq.com/freeware/freeware50/000tools/

• http://www.openvms.compaq.com/freeware/freeware50/bzip2/

• http://www.openvms.compaq.com/freeware/freeware50/info-zip/

• http://www.openvms.compaq.com/freeware/freeware40/000tools/

• ftp://ftp.process.com/vms-freeware/unzip.alpha_exe

• ftp://ftp.process.com/vms-freeware/unzip.vax_exe

• http://www.decus.de:8080/www/vms/sw/zip.htmlx

• http://www.djesys.com/zip.html

• http://www.djesys.com/unzip.html

or you can request the FILESERV_TOOLS package from the e-mail server.

Beware: The [000TOOLS...] pre-built versions of ZIP on the OpenVMS Freeware V4 (http://www.openvms.compaq.com/freeware/freeware40/) CD-ROM will erroneously return BILF errors on OpenVMS V7.2 and later. Use of the source on the Freeware V4 to rebuild the ZIP image(s), or acquiring a pre-built ZIP image from one of the above areas can avoid this. The pre-built version of ZIP on the Freeware V4 kit is older than the included ZIP sources, and it contains a latent bug.]

Directions for creaating and using the sfx self-extracting zip file compression mechanism are available in the unzip kit that is available at:

- http://www.openvms.compaq.com/freeware/freeware50/unzip542/

Specifically:

- http://www.openvms.compaq.com/freeware/freeware50/UNZIP542/UNZIPSFX.TXT

If you want to build the zip images for yourself (eg: for an older OpenVMS version), pull over the entire contents of:

- http://www.openvms.compaq.com/freeware/freeware50/unzip542/vms-binaries/

and invoke LINK.COM.

HP OpenVMS Engineering uses a tool known as FTSV for creating self-extracting compressed files using the OpenVMS DCX compression tools, as seen with various OpenVMS ECO (patch) kits. sfx provides better compression than does DCX. The FTSV and its related FTSO package have only limited availability outside HP, and are not standard products.

14 Hardware Information

14.1 What are the OpenVMS differences among VAX, Alpha, and IA-64?

In terms of software, very few. As of OpenVMS V6.1, the VAX and
Alpha platforms are very close to "feature parity". OpenVMS on IA-64 is
expected to have "feature parity" with OpenVMS Alpha, and is based on
the same source pool. Most applications can just be recompiled and run.
Some differences to be aware of:

- The default double-precision floating type on OpenVMS Alpha is VAX
  G_float, whereas on VAX it is usually D_float. D_float is available on
  Alpha, but D_float values are converted to G_float for computations
  and then converted back to D_float when stored. Because the G_float
  type has three fewer fraction bits than D_float, some applications may
  get different results. IEEE float types are also available on OpenVMS
  Alpha.

- The prefered floating point format on Alpha and IA-64 is IEEE.

- Data alignment is extremely important for best performance on Alpha.
  This means that data items should be allocated at addresses which
  are exact multiples of their sizes. Quadword alignment will offer the
  best performance, especially for character values and those smaller
  than 32 bits. Compilers will naturally align variables where they can
  and will issue warnings if they detect unaligned data items.

- HP C is the only C compiler HP offers on OpenVMS Alpha, and is
  a direct descendant of Compaq C and DEC C on OpenVMS Alpha.
  HP C is compatible with DEC C on OpenVMS VAX, but is somewhat
  different from the older VAX C compiler most people are familiar
  with. Read up on the /EXTERN_MODEL and /STANDARD qualifiers
  to avoid the most common problems. In additon to HP C, there are
  open-source ports such as Gnu C available for OpenVMS.

- The page size on Alpha and IA-64 systems is variable, but is at least
  8 kilobytes. This can have some effect on applications which use
  the $CRMPSC system service as well as on the display of available
  memory pages. The page size is available from $GETSYI(SYI$_
  PAGE_SIZE).

There are also a number of manuals which discuss migration to OpenVMS Alpha available on the documentation CD-ROM media, both in the main documentation and in the archived documentation section.

On more recent OpenVMS Alpha versions, OpenVMS Alpha has begun to add features and support not available on OpenVMS VAX. Salient new areas include the following:

- 64-bit addressing in OpenVMS Alpha V7.0 and later

- Multi-host SCSI support (SCSI TCQ) in V6.2 and later

- PCI support (platform-dependent)

- OpenVMS Galaxy support in V7.2 and later

14.2 Seeking performance information for Alpha (and VAX) systems?

HP makes a wide range of performance documents available through its FTP and WWW Internet servers (see Section 3.2).

The following contain information on current Alpha and VAX products:

- http://www.compaq.com/alphaserver/servers.html

- http://www.compaq.com/alphaserver/vax/index.html

The following sites contain information on various retired VAX and Alpha products:

- http://www.compaq.com/alphaserver/archive/index.html

- http://www.compaq.com/alphaserver/performance/perf_tps.html

Also see CPU2000:

- http://www.spec.org/osg/cpu2000/

- http://www.spec.org/osg/cpu2000/results/cpu2000.html

14.3 Where can I get updated console firmware for Alpha systems?

Firmware updates for HP Alpha systems are available from:

- ftp://ftp.digital.com/pub/Digital/Alpha/firmware/index.html

- ftp://ftp.digital.com/pub/Digital/Alpha/firmware/

- ftp://ftp.digital.com/pub/Digital/Alpha/firmware/readme.html

The latest and greatest firmware—if updated firmware has been released after the most recent firmware CD was distributed—is located at:

- ftp://ftp.digital.com/pub/Digital/Alpha/firmware/interim/

For information on creating bootable floppies containing the firmware, and for related tools, please see the following areas:

- ftp://ftp.digital.com/pub/DEC/Alpha/firmware/utilities/mkboot.txt
- ftp://ftp.digital.com/pub/DEC/Alpha/firmware/utilities/mkbootarc.txt
- ftp://ftp.digital.com/pub/DEC/Alpha/firmware/utilities/mkntboot.txt

The SROM firmware loader expects an ODS-2 formatted floppy, see mkboot. As for which image to use, the ROM image uses a header and the file extension .ROM, and the SROM bootable floppy cannot use the .ROM file.

To check the firmware loaded on recent OpenVMS Alpha systems, use the command:

```
$ write sys$output f$getsyi("console_version")
$ write sys$output f$getsyi("palcode_version")
SDA> CLUE CONFIG
```

Also see Section 14.11.

14.4 How do I boot an AlphaStation without monitor or keyboard?

The AlphaStation series will boot without a keyboard attached. To use a serial terminal as the console, issue the console command SET CONSOLE SERIAL - after that, it will use the terminal. The DEC 3000 model 300 series has a jumper on the motherboard for this purpose. Various older Alpha workstations generally will not (automatically) bootstrap without a keyboard connected, due to the self-test failure that arises when the (missing) keyboard test fails.

The usual settings for the console serial terminal (or PC terminal emulator acting as a serial console are:

```
9600 baud, 8 bits, no parity, one stop bit (9600 baud, 8N1).
```

AlphaServer 4100 and derivative series platforms, and AlphaServer GS80, GS160, and GS320 series system consoles are capable of 57600 baud. See the COM2_BAUD console environment variable, and ensure that you have current SRM firmware version loaded.

The AlphaStation and AlphaServer series use the PC DIN serial connector for the "COM1" and "COM2" serial lines, see Section 14.46 for details and pinout.

14.5   Will OpenVMS run on a Multia? AlphaPC 164LX? 164SX?

Yes, there are a set of unsupported images that permit recent OpenVMS Alpha versions to bootstrap on the Multia UDB system. These images and the associated instructions are available at the OpenVMS Freeware website:

- http://www.openvms.compaq.com/freeware/freeware50/multia/

Instructions are included IN the kits. READ THE INSTRUCTIONS. PLEASE!

Some of the restrictions involved when running OpenVMS on the Multia system include (but may well not be limited to) the following:

- The PCMCIA support was completely removed, because the Intel chip on the Multia was not compatable with the Cirrus chip on the Alphabook.
  This means, of course, that you will not see and cannot use any PCMCIA cards on a Multia.
  The Multia uses shared interrupts, and as a result, a special ZLXp-E series graphics device driver—one that does not use interrupts—is needed. This driver is provided in the kit.

- The serial lines don't work.

- If you have a Multia with a PCI slot, you can't use any PCI card that requires interrupts.

- The SRM console on this system is very old and very fragile. (This SRM console was designed only and strictly for diagnostic use, and was not particularly tested or used with OpenVMS.)

- If things don't work for you, don't expect to see any OpenVMS updates, nor SRM console updates, nor any support.

- Do not expect to see any new versions of OpenVMS on the Multia nor on any other unsupported systems. If such new versions do appear and do work, please consider it as a pleasant surprise.

The Multia images are not included on the OpenVMS Freeware V4.0 CD-ROM kit, the kit that was distributed with OpenVMS V7.2. (These images became available after Freeware V4.0 shipped.)

14–4

Other sources of information for OpenVMS on Multia include:

- http://www.djesys.com/vms/hobbyist/multia.html

- http://www.djesys.com/vms/hobbyist/mltianot.html

- http://www.djesys.com/vms/hobbyist/support.html

- http://www.netbsd.org/Ports/alpha/multiafaq.html

- http://www.brouhaha.com/~eric/computers/udb.html

OpenVMS Alpha is not supported on the AlphaPC 164LX and 164SX series, though there are folks that have gotten certain of the LX series to load SRM and bootstrap OpenVMS. (The Aspen Durango II variant.) One problem was reported: IDE bootstraps fail; SCSI is required.

Also see Section 14.10.

14.6 What is the least expensive system that will run OpenVMS?

The cheapest systems presently offered by HP that will run OpenVMS are the AlphaServer DS10 server and the AlphaStation XP900 workstation. Other companies sell Alpha-powered systems and Alpha motherboards, some of which will run (and can be purchased with) OpenVMS—see the OpenVMS Software Product Description (SPD) for details on the supported systems and configurations. There are also many used AlphaStation, AlphaServer, and DEC 3000 models available which are quite suitable. For more experienced OpenVMS system managers, the (unsupported) Multia can bootstrap OpenVMS—see Section 14.5 for details.

Depending on the OpenVMS version and configuration, the OpenVMS Software Product Description (SPD) is available at:

- http://www.openvms.compaq.com:8000/

- http://www.openvms.compaq.com/doc/

- http://www.openvms.compaq.com/commercial/

When purchasing a system, ensure that the system itself is supported, that the system disk drive is supported or closely compatible, that the CD-ROM drive is supported or is closely compatable and that (in the case of SCSI devices) it also specifically supports 512 byte block transfers; no equivalent requirement exists for IDE devices. Also particularly ensure that the video controller is supported. Use of supported HP hardware will generally reduce the level of integration effort involved.

A CD-ROM drive is required for OpenVMS Alpha installations.

CD-ROM drive compatibility information is available at:

- http://sites.inka.de/pcde/dec-cdrom-list.txt

14.7 Where can I get more information on Alpha systems?

HP operates an AlphaServer information center at:

- http://www.compaq.com/alphaserver/

Alpha Technical information and documentation is available at:

- ftp://ftp.compaq.com/pub/products/alphaCPUdocs/
- http://www.support.compaq.com/alpha-tools/documentation/current/chip-docs.html
- ftp://ftp.digital.com/pub/DEC/Alpha/systems/
- http://ftp.digital.com/pub/Digital/info/semiconductor/literature/dsc-library.html
- Alpha Systems Update:
  http://www.compaq.com/alphaserver/fb_acu.html

Software Product Description (SPD) information, including platform support documentation:

- http://www.compaq.com/info/spd/
  OpenVMS typically uses SPD 25.01.xx and/or SPD 41.87.xx.

Information on Multia hardware is available at:

- http://www.netbsd.org/Ports/alpha/multiafaq.html

Information on current and future Alpha microprocessor designs is also available from AlphaPowered at:

- http://www.alphapowered.com/alpha_tomorrow.html
- http://www.alphapowered.com/timeline.html
- http://www.alphapowered.com/ev7-and-ev8.html

The NetBSD folks maintain some Alpha hardware information at:

- http://www.netbsd.org/Ports/alpha/models.html

14.8 What are the APB boot flag values?

The following flags are passed (via register R5) to the OpenVMS Alpha primary bootstrap image APB.EXE. These flags control the particular behaviour of the bootstrap:

```
BOOT -FL root,flags

    bit      description
    ---   -----------------------------------------------
    0    CONV       Conversational bootstrap
    1    DEBUG      Load SYSTEM_DEBUG.EXE (XDELTA)
    2    INIBPT     Stop at initial system breakpoints if bit 1 set (EXEC_INIT)
    3    DIAG       Diagnostic bootstrap (loads diagboot.exe)
    4    BOOBPT     Stop at bootstrap breakpoints (APB and Sysboot)
    5    NOHEADER   Secondary bootstrap does not have an image header
    6    NOTEST     Inhibit memory test
    7    SOLICIT    Prompt for secondary bootstrap file
    8    HALT       Halt before transfer to secondary bootstrap
    9    SHADOW     Boot from shadow set
    10   ISL        LAD/LAST bootstrap
    11   PALCHECK   Disable PAL rev check halt
    12   DEBUG_BOOT  Transfer to intermediate primary bootstrap
    13   CRDFAIL       Mark CRD pages bad
    14   ALIGN_FAULTS  Report unaligned data traps in bootstrap
    15   REM_DEBUG   Allow remote high-level language debugger
    16   DBG_INIT    Enable verbose boot messages in EXEC_INIT
    17   USER_MSGS   Enable subset of verbose boot messages (user messages)
    18   RSM        Boot is controlled by RSM
    19   FOREIGN    Boot involves a "foreign" disk
```

If you want to set the boot flags "permanently" use the SET BOOT_FLAGS command, e.g.

```
>>> SET BOOT_OSFLAGS 0,1
```

14.9 What are Alpha console environment variables?

Alpha systems have a variety of variables with values set up within the SRM system console. These environment variables control the particular behaviour of the console program and the system hardware, the particular console interface presented to the operating system, various default values for the operating system bootstrap, and related control mechanisms—in other words, "the environment variables provide an easily extensible mechanism for managing complex console state."

The specific environment variables differ by platform and by firmware version—the baseline set is established by the Alpha Architecture:

```
AUTO_ACTION ("BOOT", "HALT", "RESTART", any other value
assumed to be HALT),  BOOT_DEV, BOOTDEF_DEV, BOOTED_DEV,
BOOT_FILE, BOOTED_FILE, BOOT_OSFLAGS, BOOTED_OSFLAGS,
BOOT_RESET ("ON", "OFF"), DUMP_DEV, ENABLE_AUDIT ("ON",
"OFF"), LICENSE, CHAR_SET, LANGUAGE, TTY_DEV.
```

OpenVMS Galaxy firmware can add console environment variables beginning with such strings as LP_* and HP_*, and each particular console implementation can (and often does) have various sorts of platform-specific extensions beyond these variables...

The contents of a core set of environment variables are accessable from OpenVMS using the f$getenv lexical and the sys$getenv system service. (These calls are first documented in V7.2, but have been around for quite a while.) Access to arbitary console environment variables is rather more involved, and not directly available.

14.10   Will OpenVMS run on a NoName AXPpci33?

Information on bootstrapping OpenVMS (using Multia files) on the (unsupported) NoName AXPpci33 module is available at:

- http://www.jyu.fi/~kujala/vms-in-axppci33.txt

Tips for using the Multia files with the AXPpci33:

- You have to use the Multia kit and follow the directions in ALPHA8, but do *not* load the Multia SRM firmware into the AXPpci33. Rather, download and use the latest firmware for the AXPpci33 from the HP Alpha firmware website instead.

- 64 MB memory is generally necessary.

- you cannot use any PCI cards, and if you plan on networking, you need to find an ISA Ethernet card supported by OpenVMS.

- When the AXPpci33 board bootstraps, it will dump some stuff like a crash dump, but it will continue and—so far—this hasn't caused any particular hassles.

- The system shutdown and reboot procedures do not work properly.

- The serial console is reported to not work, though the serial ports apparently do work. The status of the parallel port is unknown.

- Rumour has it that you have one of the AXPpci33 motherboards with the PS/2 mouse and keyboard connectors and a VGA card (one that will work under DECwindows) and you can run DECwindows on the system.

14.11   How do I reload SRM firmware on a half-flash Alpha system?

Some of the AlphaStation series systems are "half-flash" boxes, meaning only one set of firmware (SRM or AlphaBIOS) can be loaded in flash at a time. Getting back to the SRM firmware when AlphaBIOS (or ARC) is loaded can be a little interesting...

That said, this usually involves shuffling some files, and then getting into the AlphaBIOS firmware update sequence, and then entering "update srm" at the apu-> prompt.

To shuffle the files, copy the target SRM firmware file (as200_v7_0.exe is current) to a blank, initialized, FAT-format floppy under the filename A:\FWUPDATE.EXE

From the AlphaBIOS Setup screen, select the Upgrade AlphaBIOS option. Once the firmware update utility gets going, enter:

```
Apu-> update srm

     Answer "y" to the "Are you ready...?"
Apu-> quit
```

You've reloaded the flash. Now powercycle the box to finish the process.

Also see Section 14.3.

14.12   Will OpenVMS run on the Alpha XL series?

No. OpenVMS does not support the Alpha XL series.

OpenVMS can not, will not, and does not bootstrap on the Alpha XL series. The Alpha XL series was targeted for use (only) with the Microsoft Windows NT operating system.

For the list of boxes officially supported by OpenVMS, please see the OpenVMS Software Product Description (SPD).

- http://www.compaq.com/info/spd/
  OpenVMS typically uses SPD 25.01.xx and/or SPD 41.87.xx.

If you are very lucky, sometimes a particular unsupported Alpha box or motherboard will resemble a supported box sufficiently closely and can thus mimic that system and bootstrap. (No such family resemblances exist for the XL.) If you are exceedingly lucky, somebody here in OpenVMS Engineering will have put together a bootstrap kit—such as that for the Multia. (No Miata-like OpenVMS bootstrap kit exists for the XL.)

14.13 Describe Alpha instruction emulation and instruction subsets?

The Alpha architecture is upward- and downward-compatible, and newer instructions are emulated on older platforms, for those cases where the compiler is explicitly requested to generate the newer Alpha instructions.

In particular, OpenVMS Alpha V7.1 and later include the instruction emulation capabilities necessary for the execution of newer Alpha instructions on older Alpha microprocessors.

Alpha instructions are available in groups (or subsets). Obviously, there is the base instruction set that is available on all Alpha microprocessors. Then, the following are the current instruction extension groups (or subsets) that are available on some of various recent Alpha microprocessors:

- byte/word extension (BWX):
  LDBU, LDWU, SEXTB, SEXTW, STB, and STW.

- floating-point and square root extension (FIX):
  FTOIS, FTOIT, ITOFF, ITOFS, ITOFT, SQRTF, SQRTG, SQRTS, and SQRTT.

- count extension (CIX):
  CTLZ, CTPOP, and CTTZ.

- multi-media extension (MVI):
  MAXSB8, MAXSW4, MAXUB8, MAXUW4, MINSB8, MINSW4, MINUB8, MINUW4, PERR, PKLB, PKWB, UNPKBL, and UNPKBW.

The typical instruction subset that provides the biggest win—and of course, your mileage may vary—is typically the instruction set that is provided by the EV56 and later; specifically, the byte-word instruction subset. To select this subset, use the following:

```
/ARCHITECTURE=EV56/OPTIMIZE=TUNE=GENERIC
```

The /ARCHITECTURE controls the maximum instruction subset that the compiler will generally use, while the /OPTIMIZE=TUNE controls both the instruction-level scheduling and also the instructions generated inside loops—any code resulting from /OPTIMIZE=TUNE that is specific to an instruction subset will be generated only inside loops and will also be "protected" by an AMASK-based tesst that permits the execution of the proper code for the particular current Alpha microprocessor.

Typically /OPTIMIZE=TUNE=GENERIC is the appropriate choice for tuning, and the /ARCHITECTURE selects the minimum target architecture for general use throughout the generated code.

generated for later architectures and instruction subsets will run on older Alpha systems due to the emulation, but if /ARCHITECTURE is a significant benefit, then the emulation might be a performance penalty.

Please see the OpenVMS Ask The Wizard area for the source code of a (non-privileged) tool that looks at the instruction subsets available on the particular Alpha microprocessor that the tool is run on. This tool demonstrates the use of the Alpha AMASK and IMPLVER instructions.

14.14   What is the Accuracy of the Alpha Time of Year (BB_WATCH) Clock?

The specification for maximum clock drift in the Alpha hardware clock is 50 ppm, that's less than ±0.000050 seconds of drift per second, less than ±0.000050 days of drift per day, or less than ±0.000050 years of drift per year, etc. (eg: An error of one second over a day-long interval is roughly 11ppm, or 1000000/(24*60*60).) Put another way, this is .005%, which is around 130 seconds per month or 26 minutes per year.

The software-maintained system time can drift more, primarily due to other system activity. Typical causes of drift include extensive high-IPL code (soft memory errors, heavy activity at device IPLs, etc) that are causing the processing of the clock interrupts to be blocked.

Also see Section 14.29, Section 4.3.

14.15   So how do I open up the DEC 3000 chassis?

After removing those two little screws, tilt the back end of the top shell upwards—then you can remove the lid.

14.16 What is byte swizzling?

"Swizzling" is the term used to describe the operation needed to do partial longword (i.e. byte or word) accesses to I/O space on those systems that don't support it directly. It involved shifting the offset into an address space by 5 (or 7 for one older system), and ORing this into the base address. It then required the size of the operation to be ORed into the low order bits.

That is, because the EV4 and EV5 CPUs did not bring bits 0 and 1 off the chip, to do programmed I/O for bytes/words, the information on the size/offset of the transfer was encoded into the address data. The data itself then had to be shifted into the correct "byte lane" (i.e. its actual position within a longword).

The EV56 CPU supports the byte/word instructions however only some EV56 systems support byte/word accesses to I/O space. Even on an EV56 system that supports byte/word accesses to I/O space, the relevant OpenVMS routines do not support byte/word access to I/O space.

EV6 systems (with the exception of the AlphaServer GS60 and AlphaServer GS140 series, for reasons of platform compatability) support a flat, byte addressable I/O space.

If a device driver uses CRAM or IOC$WRITE_IO/IOC$READ_IO, then OpenVMS will do the right thing without changing the driver - OpenVMS will swizzle and unswizzle as needed.

To use byte/word operations on MEMORY, you need to tell the compiler to use the EV56 or EV6 architecture (/ARCHITECTURE=EV56). Memory operations did not swizzle, but the compiler would do long/quad access, and extract/insert bytes as needed. Using /ARCHITECTURE=EV56 allows smaller, more efficient byte/word access logic to memory.

If the application is directly doing I/O space access across a range of Alpha systems (like the graphics servers), then the driver will need to know how to do swizzling for old platforms, and byte access for new platforms.

14.17 What commands are available in the Alpha SRM console?

In addition to the normal BOOT commands and such (see Section 14.8 for some details) and the normal contents of the console HELP text, operations such as I/O redirection and floppy disk access are possible at the SRM console prompt:

1   Format a FAT floppy, and insert it into the AlphaStation floppy drive.

14–12

2   Perform the following at AlphaStation SRM Console :

```
>>> show * > env.dat
>>> show conf > conf.dat
>>> cat env.dat > fat:env.dat/dva0
>>> cat conf.dat > fat:conf.dat/dva0
```

3   You may use the SRM "ls" to display the contents of the floppy.

```
>>> ls fat:env.dat/dva0
>>> ls fat:conf.dat/dva0
```

4   You can now transfer the FAT-format floppy to another system.

14.18   How do I switch between AlphaBIOS/ARC and SRM consoles?

The specific steps required vary by system. You must first ensure that the particular Alpha system is supported by OpenVMS (see the SPD), that all core I/O components (graphics, disk controllers, etc) in the system are supported by OpenVMS (see the SPD), and that you have an OpenVMS distribution, that you have the necessary license keys (PAKs), and that you have the necessary SRM firmware loaded.

A typical sequence used for switching over from the AlphaBIOS graphics console to the SRM console follows:

1   Press F2 to get to the AlphaBIOS setup menu.

2   Pick the "CMOS Setup..." item.

3   Press F6 to get to the "Advanced CMOS Setup" menu.

4   Change the "Console Selection" to "OpenVMS Console (SRM)".

5   Press F10, F10, then Enter to save your changes.

6   Power-cycle the system.

Most Alpha systems support loading both the AlphaBIOS/ARC console and the SRM console at the same time, but systems such as the AlphaStation 255 are "half-flash" systems and do not support the presence of both the AlphaBIOS/ARC and SRM console firmware at the same time. If you have a "half-flash" system, you must load the SRM firmware from floppy, from a network download, or from a firmware CD-ROM. Following the normal AlphaBIOS or ARC firmware update sequence to the APU prompt, and then explictly select the target console. In other words, power up the system to the AlphaBIOS or ARC console, use the supplementary options to select the installation of new firmware (typically

from CD-ROM), and then rather than using a sequence which updates the current firmware:

```
Apu-> update
  -or-
Apu-> update ARC
Apu-> verify
Apu-> quit
Power-cycle the system
```

Use the following sequence to specifically update (and load) SRM from AlphaBIOS/ARC on a "half-flash" system:

```
Apu-> update SRM
Apu-> verify
Apu-> quit
Power-cycle the system
```

Use the following sequence to specifically update (and load) the AlphaBIOS/ARC console from SRM on a "half-flash" system:

```
>>> b -fl 0,A0 ddcu
BOOTFILE: firmware_boot_file.exe

Apu-> update ARC
Apu-> verify
Apu-> quit
Power-cycle the system
```

Once you have the SRM loaded, you can directly install OpenVMS or Tru64 UNIX on the system. Do not allow Windows NT to write a "harmless" signature to any disk used by OpenVMS, Tru64 UNIX, or Linux, as this will clobber a key part of the disk. (On OpenVMS, you can generally recover from this "harmless" action by using the WRITEBOOT tool.)

If you have a "full-flash" system and want to select the SRM console from the AlphaBIOS or ARC console environment, select the "Switch to OpenVMS or Tru64 UNIX console" item from the "set up the system" submenu. Then power-cycle the system. If you have a "full-flash" system with the SRM console and want to select AlphaBIOS/ARC, use the command:

```
>>> set os_type NT
```

and power-cycle the system.

For information on acquiring firmware, see Section 14.3. For information on OpenVMS license PAKs (for hobbyist use) see Section 2.7. For information on the Multia, see Section 14.5.

14–14

Information on enabling and using the failsafe firmware loader for various systems—this tool is available only on some of the various Alpha platforms—is available in the hardware documentation for the system. This tool is used/needed when the firmware has been corrupted, and cannot load new firmware.

The full list of AlphaBIOS key sequences—these sequences are needed when using an LK-series keyboard with AlphaBIOS, as AlphaBIOS expects a PC-style keyboard:

```
       F1    Ctrl/A
       F2    Ctrl/B
       F3    Ctrl/C
       F4    Ctrl/D
       F5    Ctrl/E
       F6    Ctrl/F
       F7    Ctrl/P
       F8    Ctrl/R
       F9    Ctrl/T
      F10    Ctrl/U
   Insert    Ctrl/V
   Delete    Ctrl/W
Backspace    Ctrl/H
   Escape    Ctrl/[
   Return    Ctrl/M
 LineFeed    Ctrl/J
 (Plus) +    upselect (some systems)
(Minus) -    downselect (some systems)
      TAB    down arrow
 SHIFT+TAB   up arrow
```

14.19 OpenVMS on the Personal Workstation -a and -au series?

Though OpenVMS is not supported on the Personal Workstation -a series platforms, OpenVMS might or might not bootstrap on the platform.

If you wish to attempt this, you must ensure that all graphics and all I/O controllers in the system are supported by OpenVMS. You must also ensure that you have the most current firmware loaded.

14.20 OpenVMS on the Whitebox Windows-Only series?

Though OpenVMS is not supported on the "Whitebox" series of Alpha platforms, OpenVMS might or might not bootstrap on the platform. These systems were specifically configured, targeted and supported only for use with the Microsoft Windows NT operating system.

On some of the "Whitebox" systems, the following sequence of console commands can potentially be used to convert the system over to unsupported use by and for OpenVMS Hobbyist users. (But please note that if you wish to attempt this, you must ensure that all graphics and all I/O controllers in the system are supported by OpenVMS, and you must ensure that you have the most current firmware loaded. And you must realize that the resulting configuration will be entirely unsupported.)

```
set os_type vms
cat nvram  ! too see what is in this, if anything
edit nvram
10 set srm_boot on
20 e
init
```

If your nvram has other contents, you will need to change the line numbers (10 and 20) to reflect the contents of your configuration. To obtain documentation on the commands of the console editor, enter the ? command within the editor.

The above sequence was reportedly tested on the DIGITAL Server 3300 series, a relative of the AlphaServer 800 series. The DIGITAL Server 3300 is not supported by OpenVMS, though the AlphaServer 800 series is a supported platform. The sequence may or may not work on other platforms, and may or may not work on the DIGITAL Server 3300 platform.

Also see Section 5.32.

14.21 OpenVMS and Personal Workstation IDE bootstrap?

OpenVMS will boot and is supported on the Personal Workstation -au series platforms, though OpenVMS will require a SCSI CD-ROM if the Intel Saturn I/O (SIO) IDE chip is present in the configuration— only the Cypress IDE controller chip is supported by OpenVMS for IDE bootstraps.

If you have an -au series system, you can determine which IDE chip you have using the SRM console command:

```
SHOW CONFIGURATION
```

If you see "Cypress PCI Peripheral Controller", you can bootstrap OpenVMS from IDE storage. If you see "Intel SIO 82378", you will need to use and bootstrap from SCSI. (A procedure to load DQDRIVER on the Intel SIO—once the system has bootstrapped from a SCSI device— is expected to be included as part of the contents of the DQDRIVER directory on Freeware V5.0 and later.)

14–16

Many of the -a series systems will include the Intel SIO, and thus cannot bootstrap from IDE.

14.22   Which terminal device name is assigned to the COM ports?

COM2 is normally TTA0:. COM1 is normally TTB0: if the Alpha workstation is booted with the SRM console environment variable set to graphics, and is OPA0: if the console is set to serial.

14.23   Please explain the back panel of the MicroVAX II

The MicroVAX-series console bulkhead was used with the KA630, KA650, KA655 processors.

There are three controls on the console bulkhead of these systems:

```
Triangle-in-circle-paddle: halt enable.
  dot-in-circle: halt ( break ) is enabled,
                 and auto-boot is disabled.
  dot-not-in-circle: halt ( break ) is disabled,
                 and auto-boot is enabled.

Three-position-rotary: power-up bootstrap behaviour
  arrow: normal operation.
  face: language inquiry mode.
  t-in-circle: infinite self-test loop.

Eight-position-rotary: console baud rate selection
  select the required baud rate; read at power-up.
```

There are several different bulkheads involved, including one for the BA23 and BA123 enclosures, and one for the S-box (BA2xx) series enclosure. The console bulkheads typically used either the MMJ serial line connection, or the MicroVAX DB9 (not the PC DB9 pinout), please see the descriptions of these in section WIRES1. For available adapters, see Section 14.47.

Also present on the console bulkhead is a self-test indicator: a single-digit LED display. This matches the final part of the countdown displayed on the console or workstation, and can be used by a service organization to determine the nature of a processor problem. The particular countdown sequence varies by processor type, consult the hardware or owner's manual for the processor, or contact the local hardware service organization for information the self-test sequence for a particular processor module. Note that self-tests 2, 1 and 0 are associated with the transfer of control from the console program to the (booting) operating system.

14.24   What is the layout of the VAX floating point format?

The VAX floating point format is derived from one of the PDP-11 FP formats, which helps explain its strange layout. There are four formats defined: F 32-bit single-precision, D and G 64-bit double-precision and H 128-bit quadruple precision. For all formats, the lowest addressed 16-bit "word" contains the sign and exponent (and for other than H, some of the most significant fraction bits). Each successive higher-addressed word contains the next 16 lesser-significant fraction bits. Bit 15 of the first word is the sign, 1 for negative, 0 for positive. Zero is represented by a biased exponent value of zero and a sign of zero; the fraction bits are ignored (but on Alpha, non-zero fraction bits in a zero value cause an error.) A value with biased exponent zero and sign bit 1 is a "reserved operand" - touching it causes an error - fraction bits are ignored. There are no minus zero, infinity, denormalized or NaN values.

For all formats, the fraction is normalized and the radix point assumed to be to the left of the MSB, hence $0.5 <= f < 1.0$. The MSB, always being 1, is not stored. The binary exponent is stored with a bias varying with type in bits 14:n of the lowest-addressed word.

```
  Type    Exponent bits    Exponent bias    Fraction bits (including hidden)
  =========================================================================
   F            8               128                     24
   D            8               128                     56
   G           11              1024                     53
   H           15             16384                    113
```

The layout for D is identical to that for F except for 32 additional fraction bits.

Example: +1.5 in F float is hex 000040C0 (fraction of .11[base 2], biased exponent of 129)

14.25   Where can I find more info about VAX systems?

- HP runs a VAX "InfoCenter" at:
  http://www.compaq.com/alphaserver/vax/

- Jim Agnew maintains a MicroVAX/VAXstation FAQ at:
  http://anacin.nsc.vcu.edu/~jim/mvax/mvax_faq.html

- The VAXstation 3100 Owner's Guide:
  http://www.whiteice.com/~williamwebb/intro/DOC-i.html

- A field guide to PDP-11 (and VAX) Q-bus and UNIBUS modules can be found at:
  http://metalab.unc.edu//pub/academic/computer-science/history/pdp-11/hardware/field-guide.txt

- Various VAX historical information (also see Section 2.1) can be found at:
  http://telnet.hu/hamster/vax/e_index.html

14.26 Where can I find information on NetBSD for VAX systems?

Gunnar Helliesen maintains a NetBSD VAX FAQ at http://vaxine.bitcon.no/.

14.27 What system disk size limit on the MicroVAX and VAXstation 3100?

System disks larger than 1.073 gigabytes (GB)—1fffff hexidecimal blocks – are not supported on any member of the VAXstation 3100 series and on certain older members of the MicroVAX 3100 series, and are not reliable on these affected systems. (See below to identify the affected systems— the more recent members of the MicroVAX 3100 series systems are NOT affected.)

Various of the SCSI commands used by the boot drivers imbedded in the console PROM on all members of the VAXstation 3100 series use "Group 0" commands, which allow a 21 bit block number field, which allows access to the first 1fffff hexidecimal blocks of a disk. Any disk references past 1fffff will wrap—this wrapping behaviour can be of particular interest when writing a system crashdump file, as this can potentially lead to system disk corruptions should any part of the crashdump file be located beyond 1.073 GB.

More recent systems and console PROMs use "Group 1" SCSI commands, which allow a 32 bit block number field.

There was a similar limitation among the oldest of the MicroVAX 3100 series, but a console boot PROM was phased into production and was made available for field retrofits—this PROM upgrade allows the use of the "Group 1" SCSI commands, and thus larger system disks. There was no similar PROM upgrade for the VAXstation 3100 series.

Systems that are affected by this limit:

- VAXstation 3100 series, all members. No PROM upgrade is available.

- MicroVAX 3100 models 10 and 20. No PROM upgrade is available.

14–19

- MicroVAX 3100 models 10e and 20e. Only systems with console VMB versions prior to V6.4 are affected. A PROM upgrade for these specific systems is (or was once) available.

Also see http://www.whiteice.com/~williamwebb/intro/DOC-i.html

Also see Section 9.5.

14.28   What are the VMB boot flag values?

The following flags are passed (via register R5) to the OpenVMS VAX primary bootstrap image VMB.EXE. These flags control the particular behaviour of the bootstrap:

The exact syntax is console-specific, recent VAX consoles tend to use the following:

```
>>> BOOT/R5:flags

Bit     Meaning
---     -------

 0      RPB$V_CONV
        Conversational boot. At various points in the
        system boot procedure, the bootstrap code
        solicits parameter and other input from the
        console terminal.  If the DIAG is also on then
        the diagnostic supervisor should enter "MENU"
        mode and prompt user for the devices to test.

 1      RPB$V_DEBUG
        Debug.  If this flag is set, VMS maps the code
        for the XDELTA debugger into the system page
        tables of the running system.

 2      RPB$V_INIBPT
        Initial breakpoint. If RPB$V_DEBUG is set, VMS
        executes a BPT instruction immediately after
        enabling mapping.

 3      RPB$V_BBLOCK
        Secondary boot from the boot block.  Secondary
        bootstrap is a single 512-byte block, whose LBN
        is specified in R4.

 4      RPB$V_DIAG
        Diagnostic boot.  Secondary bootstrap is image
        called [SYSMAINT]DIAGBOOT.EXE.

 5      RPB$V_BOOBPT
        Bootstrap breakpoint. Stops the primary and
        secondary bootstraps with a breakpoint
        instruction before testing memory.
```

6   RPB$V_HEADER
          Image header. Takes the transfer address of the
          secondary bootstrap image from that file's
          image header. If RPB$V_HEADER is not set,
          transfers control to the first byte of the
          secondary boot file.

7   RPB$V_NOTEST
          Memory test inhibit. Sets a bit in the PFN bit
          map for each page of memory present. Does not
          test the memory.

8   RPB$V_SOLICT
          File name. VMB prompts for the name of a
          secondary bootstrap file.

9   RPB$V_HALT
          Halt before transfer. Executes a HALT
          instruction before transferring control
          to the secondary bootstrap.

10   RPB$V_NOPFND
          No PFN deletion (not implemented; intended to
          tell VMB not to read a file from the boot device
          that identifies bad or reserved memory pages,
          so that VMB does not mark these pages as valid
          in the PFN bitmap).

11   RPB$V_MPM
          Specifies that multi-port memory is to be used
          for the total EXEC memory requirement. No local
          memory is to be used. This is for tightly-coupled
          multi-processing. If the DIAG is also on, then
          the diagnostic supervisor enters "AUTOTEST" mode.

12   RPB$V_USEMPM
          Specifies that multi-port memory should be used in
          addition to local memory, as though both were one
          single pool of pages.

13   RPB$V_MEMTEST
          Specifies that a more extensive algorithm be used
          when testing main memory for hardware
          uncorrectable (RDS) errors.

14   RPB$V_FINDMEM
          Requests use of MA780 memory if MS780 is
          insufficient for booting. Used for 11/782
          installations.

<31:28> RPB$V_TOPSYS
          Specifies the top level directory number for
          system disks with multiple systems.

14.29   What is the Accuracy of VAX the Time of Year (TOY) Clock?

The VAX Time-Of-Year (TOY) clock (used to save the time over a reboot or power failure) is specified as having an accuracy of 0.0025%. This is a drift of roughly 65 seconds per month.

The VAX Interval Time is used to keep the running time, and this has a specified accuracy of .01%. This is a drift of approximately 8.64 seconds per day. Any high-IPL activity can interfere with the IPL 22 or IPL 24 (this depends on the VAX implementation) clock interrupts—activities such as extensive device driver interrupts or memory errors are known to slow the clock.

Also see Section 14.14, Section 4.3.

14.30   Which serial port is the console on the MicroVAX 3100?

Just to keep life interesting, the MicroVAX 3100 has some "interesting" console ports behaviours based on the setting of the BREAK enable switch. When the console is not enabled to respond to BREAK, MMJ-1 is the console port. MMJ-3 will (confusingly) output the results of the selftest in parallel with MMJ-1. When the console is enabled to respond to BREAK, MMJ-3 becomes the console port, and MMJ-1 will (confusingly) output the results of selftest in parallel with MMJ-3.

14.31   How can I set up an alternate console on a VAXstation?

Most VAXstation systems have a switch—often labeled S3—that enables one of the serial lines as the system console.

Various members of the DEC 3000 series Alpha systems also have a similarly-labled S3 switch for selection of the alternate console.

Also see Section 14.4, Section 11.11, and Section 14.37.

14.32   What are the VAX processor (CPU) codes?

```
CPU:    Platform:
-----   ---------
KA41-A : MicroVAX 3100 Model 10 and 20
KA41-B : VAXserver 3100 Model 10 and 20
KA41-C : InfoServer
KA41-D : MicroVAX 3100 Model 10e and 20e
KA41-E : VAXserver 3100 Model 10e and 20e
KA42-A : VAXstation 3100 Model 30 and 40
KA42-B : VAXstation 3100 Model 38 and 48
KA43-A : VAXstation 3100 Model 76
KA45   : MicroVAX 3100 Model 30 and 40
KA46   : VAXstation 4000 Model 60
KA47   : MicroVAX 3100 Model 80
KA48   : VAXstation 4000 VLC
KA49-A : VAXstation 4000 Model 90/90A
KA49-B : VAXstation 4000 Model 95
KA49-C : VAXstation 4000 Model 96
KA50   : MicroVAX 3100 Model 90
KA51   : MicroVAX 3100 Model 95
KA52   : VAX 4000 Model 100
KA53   : VAX 4000 Model 105
KA54   : VAX 4000 Model 106
KA55   : MicroVAX 3100 Model 85
KA56   : MicroVAX 3100 Model 96
KA57   : VAX 4000 Model 108
KA58   : MicroVAX 3100 Model 88
KA59   : MicroVAX 3100 Model 98
KA85   : VAX 8500
KA86   : VAX 8600
KA88   : VAX 8800
KA600  : VAX 4000-50 (aka VAXbrick)
KA610  : MicroVAX I, VAXstation I (aka KD32)
KA620  : rtVAX (VAXeln)
KA62A  : VAX 6000-200
KA62B  : VAX 6000-300
KA630  : MicroVAX II, VAXstation II
KA640  : MicroVAX 3300, MicroVAX 3400
KA650  : VAXstation 3200, MicroVAX 3500, MicroVAX 3600, MicroVAX III
KA64A  : VAX 6000-400
KA655  : MicroVAX 3800, MicroVAX 3900, MicroVAX III+
KA65A  : VAX 6000-500
KA660  : VAX 4000-200, VAX 4 upgrade
KA66A  : VAX 6000-600
KA670  : VAX 4000-300
KA675  : VAX 4000-400
KA680  : VAX 4000-500
KA681  : VAX 4000-500A
KA690  : VAX 4000-600
KA691  : VAX 4000-605A
KA692  : VAX 4000-700A
KA693  : VAX 4000-605A
KA694  : VAX 4000-705A
```

```
KA730  : VAX-11/730
KA750  : VAX-11/750
KA780  : VAX-11/780, VAX-11/782
KA785  : VAX-11/785
KA7AA  : VAX 7000-600
KA7AB  : VAX 7000-700
KA7AC  : VAX 7000-800
KA800  : VAXrta
KA820  : VAX 8200, VAX 8300
KA825  : VAX 8250, VAX 8350
KA865  : VAX 8650
```

14.33   OpenVMS is porting to Intel IA-64?

Yes, OpenVMS is being ported to the Intel IA-64 architecture; to systems based on the Intel Itanium Processor Family.

14.34   Where can I get Intel Itanium information?

Intel Itanium Processor Family Architecture, Hardware, software, and related materials are available at:

- ftp://download.intel.com/design/IA-64/manuals/
- ftp://download.intel.com/design/IA-64/Downloads/
- ftp://download.intel.com/design/IA-64/Downloads/archSysSoftware.pdf
- ftp://download.intel.com/design/IA-64/Downloads/24870101.pdf

The Intel Extensible Firmware Interface (EFI) console documentation:

http://www.pentium.de/technology/efi/index.htm

14.35   Where can I get software and hardware support information?

Please contact the HP Customer Support Center. Services and information, manuals, guides, downloads, and various other information is available at:

- http://www.compaq.com/support/

Various hardware and system documentation is available at:

- http://www.compaq.com/support/techpubs/user_reference_guides/
- http://www.adenzel.demon.nl/vaxes/microvax3100/

- http://www.adenzel.demon.nl/vaxes/infoserver150/

TSM (Terminal Server Manager), DEChub, DECserver, etc. information:

- http://www.compaq.com/support/digital_networks_archive/

14.36 Where can I get hardware self-maintenance support assistance?

The HP Assisted Services (CAS) program (a direct descendent of the program once known as DECmailer) is available to customers that wish to maintain their own system(s) (self-maintenance), but that wish some level of assistance in acquiring hardware diagnostics and hardware manuals for the system(s), and that wish to have access to spares and module-level repairs for customer-performed hardware module swaps:

- http://www.compaq.com/CAS-Catalog/

14.37 Why does my system halt when I power-cycle the console terminal?

Various VAX and Alpha consoles are designed to process the BREAK signal, treating it as a HALT request.

A BREAK is a deliberately-generated serial line framing error.

When a serial line device such as a terminal powers up (or sometimes when powering down) it can generate framing errors. These framing errors are indistingushable from a BREAK signal.

When a BREAK is received on a serial line console for various VAX systems—including most VAXstation, MicroVAX, and VAX 4000 series—it is typically interpreted as a HALT. Alpha systems will also often process a BREAK in a similar fashion, halting the system.

There is no uniform or generally-available way to disable this behaviour on every VAX or Alpha system. On some systems, BREAK processing can be disabled in favor of [CTRL/P], or [CTRL/P] is the only way to halt the processor.

The most common way to avoid these halts is to disable the serial line console or to simply not power-cycle the console terminal. There is certain important system state information that is displayed only on the console, OpenVMS expects to always have access to the system console.

Also see Section 5.5.

14.38   Can I reuse old keyboards, mice and monitors with a PC?

Older HP keyboards (those with the DIGITAL logo and the RJ modular jacks), older HP mice (those with the DIGITAL logo and with the RJ modular jacks, or with a DIN connector with pins in a configuration other than the PC-standard DIN connector pin orientation), and older video monitors (with RGB synch-on-green video signaling) all use signaling formats and/or communications protocols that differ from the PC standards, and are not (easily) interchangable nor (easily) compatible with typical PC peripheral device controllers. LK201, LK401, VSXXX, VR260, VR290, etc., are incompatible with most PC systems.

Newer HP (and Compaq) keyboards (those with with PC-style DIN plugs, and the HP, Compaq or DIGITAL logo), newer HP mice (with PC-pin DIN plugs, and the HP, Compaq or DIGITAL logo), and newer video monitors (multi-synch) are often interchangeable with "industry standard" PC systems, and can often be used with most PC peripheral device controllers. LK461, LK471, PC7XS-CA, VRC16, VRC21, etc., are compatible with most PC systems.

Rule of thumb: if the peripheral device component was sold for use with the DEC 2000 (DECpc 150 AXP), an AlphaServer series, an AlphaStation series, or more recent Alpha system, it will probably work with a PC peripheral controller. If the peripheral device component was sold for use with an VT420 or older terminal, most VAX, most VAXstation, and most Alpha systems with names in the format 'DEC <four-digit-number>', it probably won't work on a PC.

Note that the above is a general guideline, and should not be read to indicate that any particular peripheral device will or will not work in any particular configuration, save for those specific configurations the device is explicitly supported in.

Software Integrators sells a video adapter card called Gemini P1 which will drive many of the older HP (DIGITAL-logo) fixed-frequency monitors on a PC system:

• http://www.si87.com/

The DIGITAL part number 29-32540-01 converts the output from the RGB cable (3 BNC, synch-on-green) that comes with the VAXstation 3100 and VAXstation 4000 series to a female SVGA D connector.

This will allow PC Multisync monitors with the needed frequency specifications to be used with the VAXstations. It may work with a VAXstation 2000 series, but I have not tried that combination.

14–26

The protocol definition for the old DIGITAL keyboard and mouse interfaces is buried at the back of the QDSS section in the old VAXstation II manual, specifically, in the back of the VCB02 Video Subsystem Technical Manual (EK-104AA-TM). The keyboard wiring and protocol is in appendix B, and occupies circa 44 pages. The mouse is in appendix C, circa 12 pages.

Also see Section 14.39.

14.39   Which video monitor works with which graphics controller?

To determine the answer to the "will this video monitor or this LCD panel work with this graphics controller?" question, please first locate the resolution(s) and the frequencies that are possible/supported at both ends of the video cable (on the display and on the graphics controller, in other words), and then determine if there are any matching settings available. If there are multiple matches, you will need to determine which one is most appropriate for your needs.

You will also need to determine if the video monitor or graphics controller requires the 3 BNC signaling with the synchronization signals on the green wire, or the 5 BNC signalling common on many PCs, or other connections such as the DB15 video connector or USB connector used on various systems.

If there are no matches, you will likely need to change the hardware at one or both ends of the "video cable".

The refresh frequencies for many devices have been posted to comp.os.vms and/or other newsgroups. Search the archives for details. Also see:

- http://www.repairfaq.org/

- http://www.mirage-mmc.com/faq/

- http://www.geocities.com/SiliconValley/Foothills/4467/fixedsync.html

- http://saturn.tlug.org/sunstuff/ffmonitor.html

- http://hawks.ha.md.us/hardware/monitor.html

Also see Section 14.38.

14.40   Where can I get information on storage hardware?

Information on various HP (Compaq, DIGITAL) OpenVMS and other disk storage hardware and controllers, and related technical information on SCSI, device jumpers, etc., is available at:

*   http://theref.aquascape.com/

14.41   Why does my LK401 keyboard unexpectedly autorepeat?

There are several modes of failure:

*   Pressing 2 and 3 keys at the same time causes one key to autorepeat when released. Check the hardware revision level printed on the bottom of the keyboard. If the revision level is C01, the keyboard firmware is broken. Call field service to replace the keyboard with any revision level other than C01.

*   Pressing certain keys is always broken. Typical symptoms are: delete always causes a autorepeat, return needs to be pressed twice, etc. This is frequently caused by having keys depressed while the keyboard is being initialized. Pressing ^F2 several times or unplugging and replugging the keyboard frequently fix this problem. (Ensure you have current ECO kits applied; there is a patch available to fix this problem.)

*   A key that was working spontaneously stops working correctly. This may be either of the two previous cases, or it may be bad console firmware. Ensure that you have the most recent firmware installed on your Alpha system. In particular, an old version of the DEC 3000 SRM firmware is known to have a bug that can cause this keyboard misbehaviour.

14.42   Problem - My LK411 sends the wrong keycodes or some keys are dead

Check the firmware revision on the keyboard. Hardware revision B01 introduced an incompatability with the device driver which causes the keyboard to not be recognized correctly. There is a patch available to fix this problem: [AXPDRIV06_061] - the fix is also included in OpenVMS V6.2. The rev A01 keyboard, and the LK450 should work without problems.

If you are working from another operating system platform, please see the DECxterm tool and related information on OpenVMS Freeware V5.0.

14.43 Which DE500 variant works with which OpenVMS version?

Ensure you have a version of the Alpha SRM console with support for the DE500 series device. Apply ALL mandatory ECO kits for the OpenVMS version in use, and also apply the CLUSIO, ALPBOOT, and ALPLAN kits, and apply any available ALPCPU ECO kit for the platform.

- DE500-XA
  auto-detection, no auto-negotiation,
  OpenVMS V6.2-1H1 and ALPBOOT ECO, also V7.0 and later and ECO.
  Device hardware id 02000011 and 02000012.
  Component part number 54-24187-01

- DE500-AA
  auto-detection, auto-negotiation,
  OpenVMS V6.2 and ALPBOOT and ALPLAN ECOs, or V7.1 and later and ECO.
  Device hardware id 02000020 and 20000022.
  Component part number 54-24502-01

- DE500-BA
  auto-detection, auto-negotiation,
  OpenVMS V6.2-1H3 and CLUSIO, ALPBOOT, ALPLAN and ALPCPU ECOs, or V7.1-1H1 or later and ECO.
  Device hardware id 02000030 (check connector, vs DE500-FA) (other values on old Alpha SRM firmware)
  Component part number 54-24602-01

- DE500-FA (100 megabit fibre optic Ethernet)
  OpenVMS V7.1-1H1 and later
  Device hardware id 02000030 (check connector, vs DE500-BA) (other values possible on old Alpha SRM firmware)
  Component part number 54-24899-01

To check the DE500 device hardware id from OpenVMS, use the following command:

```
$ ANALYZE/SYSTEM
SDA> SHOW LAN/DEVICE=EWcu:
```

The "hardware id" will be displayed.

To set the DE500 speed via the Alpha SRM console environment variable:

```
EWx0_MODE setting            Meaning
------------------------     --------------------------------
Twisted-Pair                 10 Mbit/sec, nofull_duplex
Full Duplex, Twisted-Pair    10 Mbit/sec, full_duplex
AUI                          10 Mbit/sec, nofull_duplex
BNC                          10 Mbit/sec, nofull_duplex
Fast                         100 Mbit/sec, nofull_duplex
FastFD (Full Duplex)         100 Mbit/sec, full_duplex
Auto-Negotiate               Negotiation with remote device
```

To override the console setting and use LANCP:

```
$ RUN SYS$SYSTEM:LANCP
LANCP> SET DEV EWA0/SPEED=10
LANCP> SET DEV EWA0/SPEED=100/full_duplex
```

Fast Ethernet (100Base, 100 megabit) controllers such as the DE500 series have a pair of connections available—while traditional Ethernet (10Base, 10 megabit) is inherently a half-duplex protocol, Fast Ethernet can be configured to use one or both of the available connections, depending on the controller. Fast Ethernet can thus be half- or full-duplex depending on the configuration and the capabilities of the network controller and the Ethernet network plant. Some Fast Ethernet controllers can also operate at traditional Ethernet speeds, these controllers are thus often refered to as 10/100 Ethernet controllers.

14.44   Third-party disk/tape/controllers/SCSI/widgets on OpenVMS?

A wide variety of third-party widgets—SCSI and IDE disks and tapes, graphics controllers, etc—are available for various platforms.

If you purchase third-party "generic" SCSI or IDE storage devices, you and your device vendor will be responsible for the testing and the support of the devices. In general, you can expect that Compaq will address non-standards-compliance problems within OpenVMS (changes that will also not prevent operations with other supported devices, of course), but you and/or the device vendor and/or the device manufacturer are responsible for finding and fixing problems in the particular third-party device and or controller involved.

In particular, realize that neither SCSI nor IDE is a particularly standard interface, these interfaces tend to be a collection of optionally-implemented and standardized interface features. You should not and can not simply assume that all SCSI nor IDE storage devices are interchangeable. If you want to try to use a generic SCSI device, use

V6.2 or later, or (better) V7.1-2 or later. If you wish to try to use IDE, use OpenVMS V7.1-2 or later.

On older OpenVMS releases, see the disk capacity limits (Section 9.5).

With SCSI disks on releases prior to V6.2, ensure that you have the ARRE and ARWE settings configured correctly (disabled). (If not, you will see DRVERR fatal drive errors and error log entries.)

Some SCSI disks set the medium type byte as part of the SCSI size field—this is a SET CAPACITY extension to SCSI specs. This problem also applies to VAX V7.1 and later.

Disks with SCSI disk sizes past 8.58 GB and/or with the SET CAPACITY extension require ALPSCSI07 ECO or the OpenVMS Alpha V7.1-2 or later release. (See Section 9.5 for further details.)

Based on the displays of the (undocumented) SYS$ETC:SCSI_INFO tool; this tool is present in OpenVMS V6.2 and later:

```
Issuing 6-byte MODE SENSE QIOW to get current values for page 01h
        Page Code .................. 01h
        Page Name .................. Read-Write Error Recovery
        Saveable ................... Yes
        Size ....................... 10
        Hex Data ................... E6 08 50 00 00 00 08 00
                                     00 00
```

The E6 indicates that the AWRE and ARRE bits are set, and this is not acceptable on OpenVMS versions prior to V6.2. Further along in the SCSI_INFO display, if you also see:

```
Issuing 6-byte MODE SENSE QIOW to get changeable values for page 81h
        Page Code .................. 01h
        Page Name .................. Read-Write Error Recovery
        Saveable ................... Yes
        Size ....................... 10
        Hex Data ................... C0 08 50 00 00 00 08 00
                                     00 00
```

The C0 value means that the AWRE and ARRE values can be changed on this particular SCSI device. (This is not always the case.) Use RZDISK from the OpenVMS Freeware, and reset the E6 flag byte to hexadecimal 26 (or whatever the remaining mask when you remove bits C0) on page one.

Each SCSI and IDE host contains non-trivial SCSI and IDE driver software, and each device contains equally non-trivial firmware— taken together with the mechanical and electronic components, this software and firmware will determine whether or not a particular device will function as expected.

Also note that various devices—such as various SCSI CD-R devices — can implement and can require vendor-specific protocol extensions, and these extensions can require modifications to OpenVMS or the addition of various utilities. In various of these cases, these devices perform functions that will require them to use SCSI or IDE commands that are (hopefully) architecturally-compatible SCSI or IDE command extensions. (Also see Section 7.1 and Section 9.7.)

In order for OpenVMS to officially support a particular device, integration and testing work is mandated. There can be no certainty that any particular device will operate as expected in any particular configuration without first performing this (non-trivial) work.

It is quite possible to find two devices—both entirely compliant with applicable standards or interface documents—that will not interoperate.

The same general statement holds for OpenVMS bootstrapping on an unsupported VAX or Alpha platform. It might or might not work. In particular, please see the OpenVMS Software Product Description (SPD) for the list of platforms supported by OpenVMS. OpenVMS is not supported on the Personal Workstation -a series, on the Digital Server series platforms, on the AlphaServer 2100 series 5/375 CPU, on the Multia, on the AlphaServer DS20L, and on a variety of other platforms. (You might or might not see success booting OpenVMS on any of these platforms.)

14.44.1   Lists of third-party widgets on OpenVMS?

Various folks have successfully used common third-party disk disk devices with OpenVMS, such as the IDE and SCSI variants of the Iomega Zip250 removable disk device. Common SCSI CD-R/CD-RW devices such as the Plextor PlexWriter 12/10/32S SCSI series have also been successfully utilized with various AlphaStation and VAXstation systems, and with tools such as CDRECORD. (A Plextor burn of 614400000 bytes (300000 sectors) requires just over six minutes at 12x, using an AlphaStation XP1000 666 MHz EV67 system UltraSCSI host.)

If you choose to attempt to use third-party devices, ensure that you have the current OpenVMS version and the current ECO kit(s) applied. In the case of the IDE Zip250 drive, ensure that you have the most current revision of SYS$DQDRIVER installed.

14.44.2   Are the 2X-KZPCA-AA and SN-KZPCA-AA LVD Ultra2 SCSI?

Yes. Both of these controllers are Ultra2 low-voltage differential (LVD) SCSI controllers.

14.44.3   Resolving DRVERR fatal device error?

If this is on an OpenVMS version prior to V6.2, please see the AWRE and ARRE information included in section Section 14.44.

14.45   How do I convert? Disk Blocks? KB, MB, GB, TB?

The granularity of disk storage allocation is called a disk block. Groups of disk blocks are organized together into the smallest unit of storage that can be allocated, and this unit is called a disk cluster. The number of blocks in a cluster is the cluster factor, and is established when the disk volume is initialized.

Each individual disk block is composed of five hundred twelve (512) bytes, or one-half kilobyte. Each byte is comprised of eight bits. A bit represents the smallest unit of information, typically refered to as a one or a zero.

OpenVMS tends to uses base two notation for disk storage, while disk storage capacity specifications from most storage vendors (including Compaq) will generally use base ten notation.

An OpenVMS disk block is 512 bytes in size; this is one-half kilobyte in base two notation.

The following table describes the prefix, the abbreviation, and the associated base ten (marketing) and base two (OpenVMS) values.

```
          Base Ten                           Base Two
          -----------------------------    ----------------------
Kilobyte  (KB)  10**3               1000   2**10             1024
Megabyte  (MB)  10**6            1000000   2**20          1048576
Gigabyte  (GB)  10**9         1000000000   2**30       1073741824
Terabyte  (TB)  10**12     1000000000000   2**40     1099511627776
Petabyte  (PB)  10**15  1000000000000000   2**50  1125899906842624
```

The base ten representation of the 2**40 value is 1099511627776, which is obviously rather ugly. When viewed as a base eight or base sixteen (octal or hexadecimal, respectively) value, the value is far nicer. Specifically, the value is 10000000000 and 40000000 when represented in octal and hexadecimal, respectively.

Notational note: Within the OpenVMS FAQ, a Kilobit is represented by the appreviation Kb, while a Kilobyte is represented as KB.

To convert OpenVMS disk blocks to (base two) kilobytes (KB; 1024 bytes), simply divide by two. To convert blocks to (base two) megabytes, divide by 2048. Blocks to (base two) gigabytes (GB), divide by 2097152. These particular divisions can also be performed using bitshifts: to divide a value by two, shift the binary value rightwards by one bit position.

To convert OpenVMS disk blocks to (base ten) kilobytes, divide by approximately 1.953125.

And for those rummaging around deep in SYSGEN, a microfortnight is approximately one second.

14.46   Looking for connector wiring pinouts?

```
The DECconnect DEC-423 Modified Modular Jack (MMJ) pinout:

  1: Data Terminal Ready (DTR)
  2: Transmit (TXD)
  3: Transmit Ground (TXD-)
  4: Receive Ground (RXD-)
  5: Receive (RXD)
  6: Data Set Ready (DSR)

   +------------------+
   | 1  2  3  4  5  6 |
   +-----------+    ++
              +____+
```

The PC-compatible DB9 connector pinout follows:

```
  1: Data Carrier Detect (DCD)
  2: Received Data
  3: Transmit Data
  4: Data Terminal Ready (DTR)
  5: Ground
  6: Data Set Ready (DSR)
  7: Request To Send (RTS)
  8: Clear To Send
  9: floating
```

The MicroVAX DB9 console connector pinout predates the PC-style DB9 pinout, and uses a then-common (older) standard pinout, and uses the following EIA-232 series standard signals:

```
1: Protective Ground
2: Transmited Data
3: Received Data
4: Request To Send (RTS)
5: Data Terminal Ready (DTR)
6: Data Set Ready (DSR)
7: Signal Ground
8: Shorted to pin 9 on MicroVAX and VAXstation 2000...
9:    ...series systems, otherwise left floating.

When pin 8 is shorted to pin 9, this is a BCC08 (or variant) cable,
most commonly used as a console cable on the MicroVAX 2000 and
VAXstation 2000 series.  (Other systems may or may not tolerate
connecting pin 8 to pin 9.)
```

The BC16E-nn (where -nn indicates the cable length) cable key impliicitly "flips over" (crosses-over) the signal wires, so all DECconnect MMJ connectors are wired the same.

```
  //
  ----                                        ----
 |  |-------------------------------|  |
  ----                                        ----
                                               \\
```

The BC16-E-nn cross-over wiring looks like this:

```
        Terminal                        Host
        MMJ                             MMJ

    DTR 1 --->---------->----------->--- 6 DSR
    TXD 2 --->---------->----------->--- 5 RXD
        3 ----------------------------- 4
        4 ----------------------------- 3
    RXD 5 ---<----------<-----------<--- 2 TXD
    DSR 6 ---<----------<-----------<--- 1 DTR
```

The BN24H looks like this:

```
    MMJ         RJ45

     1---------8
     2---------2
     3---------1
     4---------3
     5---------6
     6---------7
```

The BN24J looks like this:

```
MMJ        RJ45

 1---------7
 2---------6
 3---------3
 4---------1
 5---------2
 6---------8
```

Also see:

- http://www.openvms.compaq.com/wizard/padapters.html

- http://www.airborn.com.au/rs232.html

- http://www.stanq.com/cable.html

- For adapters and connectors, see Section 14.47.

14.47   What connectors and wiring adapters are available?

The H8571-B converts the (non-2000-series) MicroVAX DB9 to MMJ
DECconnect. The MicroVAX 2000 and VAXstation 2000 requires a BCC08
cable (which has the 8-9 short, see Section 14.46) and the H8571-D for
use with DECconnect.

More recent HP (HP, Compaq or DIGITAL logo) systems will use either
the DECconnect MMJ wiring or (on all recent system designs) the PC-
compatible DB9 pinout.

DECconnect MMJ adapters:

```
    Part:      Converts BC16E MMJ male to fit into:

    H8571-C  25 pin DSUB Female to MMJ, Unfiltered
    H8571-D  EIA232 25 pin male (modem-wired)
    H8571-E  25 pin DSUB Female to MMJ, Filtered
    H8571-J  PC/AT 9 pin male (PC serial port)
    H8572-0  BC16E MMJ double-female (MMJ extender)
    H8575-A  EIA232 25 pin female (common)
    H8575-B  EIA232 9 pin male (MicroVAX II console)
    H8575-D  25 Pin to MMJ W/EOS and ESD Protection
    H8577-AA 6 pin Female MMJ to 8 pin MJ
    BC16E-** MMJ cable, available in various lengths
```

Numerous additional adapters and cables are available from the _OPEN
DECconnect Building Wiring Components and Applications Catalog_, as
well as descriptions of the above-listed parts.

The H8571-A and H8575-A are MMJ to DB25 (female) and are wired as follows:

Also see:

*   http://www.openvms.compaq.com/wizard/padapters.html

Jameco offers a USB-A to PS/2 Mini DIN 6 Adapter (as part 168751), for those folks wishing to (try to) use PS/2 Keyboards via USB-A connections.

14.48 What is flow control and how does it work?

XON/XOFF is one kind of flow control.

In ASCII, XON is the `CTRL/Q` character, and XOFF is the `CTRL/S`.

XON/XOFF flow control is typically associated with asynchronous serial line communications. XON/XOFF is an in-band flow control, meaning that the flow control is mixed in with the data.

CTS/RTS is another type of flow control, and is sometimes called hardware flow control. Out-of-band means that seperate lines/pins from the data lines (pins) are used to carry the CTS/RTS signals.

Both kinds of flow control are triggered when a threshold is reached in the incoming buffer. The flow control is suppose to reach the transmitter in time to have it stop transmitting before the receiver buffer is full and data is lost. Later, after a sufficient amount of the receiver's buffer is freed up, the resume flow control signal is sent to get the transmitter going again.

DECnet Phase IV on OpenVMS VAX supports the use of asynchronous serial communications as a network line. The communication devices (eg. modems, and drivers) *must not* be configured for XON/XOFF flow control. The incidence of these (unexpected) in-band characters will corrupt data packets. Further, the serial line device drivers might normally remove the XON and XOFF characters from the stream for terminal applications, but DECnet configures the driver to pass *all* characters through and requires that all characters be permitted. (The communication devices must pass through not only the XON and XOFF characters, they must pass *all* characters including the 8-bit characters. If data compression is happening, it must reproduce the source stream exactly. No addition or elimination of null characters, and full data transparency.

An Ethernet network is rather different than an asynchronous serial line. Ethernet specifies the control of data flow on a shared segment using CSMA/CD (Carrier Sense Multiple Access, with Collision Detect) An Ethernet station that is ready to transmit listens for a clear channel (Carrier Sense). When the channel is clear, the station begins to transmit by asserting a carrier and encoding the packet appropriately. The station concurrently listens to its own signal, to permit the station to detect if another station began to transmit at the same time—this is called collision detection. (The collision corrupts the signal in a way that can reliably be detected.) Upon detecting the collision, both stations will stop transmitting, and will back off and try again a little later. (You can see a log of this activity in the DECnet NCP network counters.)

DECnet provides its own flow control, above and beyond the flow control of the physical layer (if any). The end nodes handshake at the beginning to establish a transmit window size—and a transmitter will only send that much data before stopping and waiting for an acknowledgement. The acknowledgement is only sent when the receiver has confirmed the packet is valid. (A well-configured DECnet generally avoids triggering any underlying (out-of-band) flow control mechanism.)

15 Information on Networks and Clusters

The following sections contain information on OpenVMS Networking with IP and DECnet, and on clustering and volume shadowing, on Fibre Channel, and on related products and configurations.

15.1 How to connect OpenVMS to a Modem?

Please see the Ask The Wizard area topics starting with (81), (1839), (2177), (3605), etc.

- http://www.openvms.compaq.com/wizard/
- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

15.2 OpenVMS and IP Networking?

The following sections contain information on OpenVMS and IP networking, as well as IP printing topics.

15.2.1 How to connect OpenVMS to the Internet?

Some tutorial information and tips for connecting OpenVMS systems to the Internet are available at:

- http://www.tmesis.com/internet/

15.2.2 Connecting to an IP Printer?

To connect a printer via the IP telnet or lpr/lpd protocols, you will need to install and configure an IP stack on OpenVMS, and configure the appropriate print queue.

With current OpenVMS IP implementations, the choice of telnet or lpr/lpd really amounts to determining which of these works better with the particular printer involved.

please see the Ask The Wizard area topics starting with (1020) for additional information.

- http://www.openvms.compaq.com/wizard/

• http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

Please see Section 15.2.3 for information on Postscript printing.
comment>(———————————————————————)

15.2.3 How do I connect a PostScript printer via TCP/IP?

Using UCX as the TCP/IP stack, it is possible to setup queues using the
UCX$TELNETSYM in order to print to postscript printers. This assumes
however that the printer itself can convert whatever is passed to it into
something intelligible. As an example, if the printer has an IP address of
123.456.789.101 and jobs should be passed to port 9100 then :

```
$ INITIALIZE/QUEUE/ON="123.456.789.101:9100" -
    /PROCESSOR=UCX$TELNETSYM  -
    my_ip_queue
```

The port number of 9100 is typical of HP JetDirect cards but may be
different for other manufacturers cards.

As a better alternative, DCPS Version 1.4 and later support IP queues
using either HP TCP/IP Services for OpenVMS software or Cisco Multinet
for OpenVMS. The usage of this type of interface is documented in the
Release Notes and the DCPS$STARTUP.TEMPLATE file.

For general and additional (non-Postscript) IP printing information,
please see topic (1020) and other topics referenced in that topic in the Ask
The Wizard area.

• http://www.openvms.compaq.com/wizard/

• http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9. Also see:

• http://www.openvms.compaq.com/wizard/

• http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9. Also see:

• http://www.wotsit.org/

p>Please see Section 15.2.2 for pointers to an introduction to IP printing.

15.2.4 How do I set a default IP route or gateway on OpenVMS?

If you have TCP/IP Services, then use the command for TCP/IP Services V5.0 and later:

```
$ TCPIP SET ROUTE/GATE=x.x.x.x/DEFAULT/PERMANENT
```

And for earlier TCP/IP Services versions, use the command:

```
$ UCX SET ROUTE/GATE=x.x.x.x/DEFAULT/PERMANENT
```

15.2.5 How can I set up reverse telnet (like reverse LAT)?

Though it may seem obvious, Telnet and LAT are quite different—with differing capabilities and design goals.

Please see the documentation around the TCP/IP Services for OpenVMS TELNET command CREATE_SESSION. This command is the equivilent of the operations performed in LTLOAD.COM or LAT$SYSTARTUP.COM. There is no TELNET equivilent to the sys$qio[w] control interface for LTDRIVER (as documented in the I/O User's Reference Manual) available, though standard sys$qio[w] calls referencing the created TN device would likely operate as expected.

15.2.6 Why can't I use PPP and RAS to connect to OpenVMS Alpha?

OpenVMS Alpha IP PPP does not presently support authentication, and the Microsoft Windows NT option to disable authentication during a RAS connection apparently doesn't currently work—RAS connections will require authentication—and this will thus prevent RAS connections.

Future versions of OpenVMS and TCP/IP Services may add this, and future versions of Microsoft Windows may permit operations with authentication disabled.

15.3 OpenVMS and DECnet Networking?

The following sections contain information on OpenVMS and DECnet networking.

15.3.1 Can DECnet-Plus operate over IP?

Yes. To configure DECnet-Plus to operate over IP transport and over IP backbone networks, install and configure DECnet-Plus, and install and configure the PWIP mechanism within the IP stack.

15.3.2 What does "failure on back translate address request" mean?

The error message:

```
BCKTRNSFAIL, failure on the back translate address request
```

indicates that the destination node is running DECnet-Plus, and that its naming service (DECnet-Plus DECdns, LOCAL node database, etc) cannot locate a name to associate with the source node's address. In other words, the destination node cannot determine the node name for the node that is the source of the incoming connection.

Use the DECNET_REGISTER mechanism (on the destination node) to register or modify the name(s) and the address(es) of the source node. Check the namespace on the source node, as well.

Typically, the nodes involved are using a LOCAL namespace, and the node name and address settings are not coherent across all nodes. Also check to make sure that the node is entered into its own LOCAL namespace. This can be a problem elsewhere, however. Very rarely, a cache corruption has been known to cause this error. To flush the cache, use the command:

```
$ RUN SYS$SYSTEM:NCL
flush session control naming cache entry "*"
```

Also check to see that you are using the latest ECO for DECnet-Plus for the version you are running. DECnet-Plus can use the following namespaces:

- DECdns: DECnet-Plus distributed name services.

- LocalFile: a local file containing names and addresses.

- DNS/BIND: the IP distributed name services.

- The UCX local host file.

Of these, searching DNS/BIND and LocalFile, respectively, is often the most appropriate configuration.

15.3.3   Performing SET HOST/MOP in DECnet-Plus?

First, do MCR NCL SHOW MOP CIRCUIT *

Let us assume you have a circuit known as FDDI-0. Here is an example
of the SET HOST/MOP command syntax:

```
$ SET HOST/MOP/ADDRESS=08-00-2B-2C-5A-23/CIRCUIT=FDDI-0
```

Also see Section 15.6.3.

15.4   How to determine the network hardware address?

Most Alpha and most VAX systems have a console command that displays
the network hardware address. Many systems will also have a sticker
identifying the address, either on the enclosure or on the network
controller itself.

The system console power-up messages on a number of VAX and Alpha
systems will display the hardware address, particularly on those systems
with an integrated Ethernet network adapter present.

If you cannot locate a sticker on the system, if the system powerup
message is unavailable or does not display the address, and if the system
is at the console prompt, start with the console command:

```
HELP
```

A console command similar to one of the following is typically used to
display the hardware address:

```
SHOW DEVICE
SHOW ETHER
SHOW CONFIG
```

On the oldest VAX Q-bus systems, the following console command can
be used to read the address directly off the (DELQA, DESQA, or the
not-supported-in-V5.5-and-later DEQNA) Ethernet controller:

```
E/P/W/N:5 20001920
```

Look at the low byte of the six words displayed by the above command.
(The oldest VAX Q-bus systems—such as the KA630 processor module
used on the MicroVAX II and VAXstation II series—lack a console
HELP command, and these systems typically have the primary network
controller installed such that the hardware address value is located at the
system physical address 20001920.)

If the system is a VAX system, and another VAX system on the network is configured to answer Maintenance and Operations Protocol (MOP) bootstrap requests (via DECnet Phase IV, DECnet-Plus, or LANCP), the MOM$SYSTEM:READ_ADDR.EXE tool can be requested:

```
B/R5:100 ddcu
Bootfile: READ_ADDR
```

Where ddcu is the name of the Ethernet controller in the above command. The primarly local DELQA, DESQA, and DEQNA Q-bus controllers are usually named XQA0. An attempt to MOP download the READ_ADDR program will ensue, and (if the download is successful) READ_ADDR will display the hardware address.

If the system is running, you can use DECnet or TCP/IP to display the hardware address with one of the following commands.

```
$ MCR NCP SHOW KNOWN LINE CHARACTERISTICS     ! DECnet Phase IV

$ MCR NCL SHOW CSMA-CD STATION * ALL STATUS  ! DECnet-Plus

$ UCX SHOW INTERFACE/FULL    ! TCP/IP versions prior to V5.0

$ TCPIP SHOW INTERFACE/FULL  ! TCP/IP versions V5.0 and later
```

A program can be created to display the hardware address, reading the necessary information from the network device drivers. An example C program for reading the Ethernet hardware address (via sys$qio calls to the network device driver(s)) is available at the following URL:

• http://www.openvms.compaq.com/wizard/swdev/ethernVMS.html

To use the DECnet Phase IV configurator tool to watch for MOP SYSID activity on the local area network:

```
$ NCP SET MODULE CONFIGURATOR KNOWN CIRCUIT SURVEILLANCE ENABLED
```

Let the DECnet configurator run for at least 20 minutes. Then issue the following commands:

```
$ NCP SHOW MODULE CONFIGURATOR KNOWN CIRCUIT STATUS TO filename.txt
$ NCP SET MODULE CONFIGURATOR KNOWN CIRCUIT SURVEILLANCE DISABLED
```

The resulting file (named filename.txt) can now be searched for the information of interest. Most DECnet systems will generate MOP SYSID messages identifying items such as the controller hardware address and the controller type, and these messages are generated and multicast roughly every ten minutes.

Information on the DECnet MOP SYSID messages and other parts of the maintenance protocols is included in the DECnet network architecture specifications referenced in section DOC9.

15.4.1 How do I reset the LAN (DECnet-Plus NCL) error counters?

On recent OpenVMS releases:

```
$ RUN SYS$SYSTEM:LANCP
SET DEVICE/DEVICE_SPECIFIC=FUNCTION="CCOU" devname
```

15.4.2 How do I install DECnet Phase IV on VMS 7.1?

On OpenVMS V7.1, all DECnet binaries were relocated into separate installation kits—you can selectively install the appropriate network: DECnet-Plus (formerly known as DECnet OSI), DECnet Phase IV, and HP TCP/IP Services (often known as UCX).

On OpenVMS versions prior to V7.1, DECnet Phase IV was integrated, and there was no installation question. You had to install the DECnet-Plus (DECnet/OSI) package on the system, after the OpenVMS upgrade or installation completed.

During an OpenVMS V7.1 installation or upgrade, the installation procedure will query you to learn if DECnet-Plus should be installed. If you are upgrading to V7.1 from an earlier release or are installing V7.1 from a distribution kit, simply answer "NO" to the question asking you if you want DECnet-Plus. Then—after the OpenVMS upgrade or installation completes – use the PCSI PRODUCT INSTALL command to install the DECnet Phase IV binaries from the kit provided on the OpenVMS software distribution kit.

If you already have DECnet-Plus installed and wish to revert, you must reconfigure OpenVMS. You cannot reconfigure the "live" system, hence you must reboot the system using the V7.1 distribution CD-ROM. Then select the DCL ($$$ prompt) option. Then issue the commands:

```
$$$ DEFINE/SYSTEM PCSI$SYSDEVICE DKA0:
$$$ DEFINE/SYSTEM PCSI$SPECIFIC DKA0:[SYS0.]
$$$ PRODUCT RECONFIGURE VMS /REMOTE/SOURCE=DKA0:[VMS$COMMON]
```

The above commands assume that the target system device and system root are "DKA0:[SYS0.]". Replace this with the actual target device and root, as appropriate. The RECONFIGURE command will then issue a series of prompts. You will want to reconfigure DECnet-Plus off the system, obviously. You will then want to use the PCSI command

15–7

PRODUCT INSTALL to install the DECnet Phase IV kit from the OpenVMS distribution media.

Information on DECnet support, and on the kit names, is included in the OpenVMS V7.1 installation and upgrade documentation.

Subsequent OpenVMS upgrade and installation procedures can and do offer both DECnet Phase IV and DECnet-Plus installations.

15.5 How can I send radio-frequency pages from my OpenVMS system?

There are third-party products available to send messages to radio paging devices (pagers), communicating via various protocols such as TAP (Telocator Alphanumeric Protocol).

RamPage (Ergonomic Solutions) is one of the available packages that can generate and transmit messages to radio pagers. Target Alert (Target Systems; formerly the DECalert product) is another. Networking Dynamics Corp has a product called Pager Plus. The System Watchdog package can also send pages. The Process Software package PMDF can route specific email addresses to a paging service, as well.

Many commercial paging services provide email contact addresses for their paging customers—you can simply send email directly to the pager.

Some people implement the sending of pages to radio pagers by sending commands to a modem to take the "phone" off the "hook", and then the paging sequence, followed by a delay, and then the same number that a human would dial to send a numeric page. (This is not entirely reliable, as the modem lacks "call progress detection", and the program could simply send the dial sequence when not really connected to the paging company's telephone-based dial-up receiver.)

See Section 13.1 for information on the available catalog of products.

15.6 OpenVMS, Clusters, Volume Shadowing?

The following sections contain information on OpenVMS and Clusters, Volume Shadowing, and Cluster-related system parameters.

15.6.1 OpenVMS Cluster Communications Protocol Details?

The following sections contain information on the OpenVMS System Communications Services (SCS) Protocol.

15.6.1.1 OpenVMS Cluster (SCS) over DECnet? Over IP?

The OpenVMS Cluster environment operates over various network protocols, but the core of clustering uses the System Communications Services (SCS) protocols, and SCS-specific network datagrams. Direct (full) connectivity is assumed.

An OpenVMS Cluster DOES NOT operate over DECnet, nor over IP.

No SCS protocol routers are available.

Many folks have suggested operating SCS over DECnet or IP over the years, but SCS is too far down in the layers, and any such project would entail a major or complete rewrite of SCS and of the DECnet or IP drivers. Further, the current DECnet and IP implementations have large tracts of code that operate at the application level, while SCS must operate in the rather more primitive contexts of the system and particularly the bootstrap—to get SCS to operate over a DECnet or IP connection would require relocating major portions of the DECnet or IP stack into the kernel. (And it is not clear that the result would even meet the bandwidth and latency expectations.)

The usual approach for multi-site OpenVMS Cluster configurations involves FDDI, Memory Channel (MC2), or a point-to-point remote bridge, brouter, or switch. The connection must be transparent, and it must operate at 10 megabits per second or better (Ethernet speed), with latency characteristics similar to that of Ethernet or better. Various sites use FDDI, MC2, ATM, or point-to-point T3 link.

15.6.1.2 Configuring Cluster SCS for path load balancing?

SCS: Systems Communication Services. The protocol used to communicate between VMSCluster systems and between OpenVMS systems and SCS-based storage controllers. (SCSI-based storage controllers do not use SCS.)

PORT: A communications device, such as DSSI, CI, Ethernet or FDDI. Each CI or DSSI bus is a different local port, named PAA0, PAB0, PAC0 etc. All Ethernet and FDDI busses make up a single PEA0 port.

VIRTUAL CIRCUIT: A reliable communications path established between a pair of ports. Each port in a VMScluster establishes a virtual circuit with every other port in that cluster.

All systems and storage controllers establish "Virtual Circuits" to enable communications between all available pairs of ports.

SYSAP: A "system application" that communicates using SCS. Each SYSAP communicates with a particular remote SYSAP. Example SYSAPs include:

VMS$DISK_CL_DRIVER connects to MSCP$DISK
The disk class driver is on every VMSCluster system. MSCP$DISK is on all disk controllers and all VMSCluster systems that have SYSGEN parameter MSCP_LOAD set to 1

VMS$TAPE_CL_DRIVER connects to MSCP$TAPE
The tape class driver is on every VMSCluster system. MSCP$TAPE is on all tape controllers and all VMSCluster systems that have SYSGEN parameter TMSCP_LOAD set to 1

VMS$VAXCLUSTER connects to VMS$VAXCLUSTER
This SYSAP contains the connection manager, which manages cluster connectivity, runs the cluster state transition algorithm, and implements the cluster quorum algorithm. This SYSAP also handles lock traffic, and various other cluster communications functions.

SCS$DIR_LOOKUP connects to SCS$DIRECTORY
This SYSAP is used to find SYSAPs on remote systems

MSCP and TMSCP
The Mass Storage Control Protocol and the Tape MSCP servers are SYSAPs that provide access to disk and tape storage, typically operating over SCS protocols. MSCP and TMSCP SYSAPs exist within OpenVMS (for OpenVMS hosts serving disks and tapes), within CI- and DSSI-based storage controllers, and within host-based MSCP- or TMSCP storage controllers. MSCP and TMSCP can be used to serve MSCP and TMSCP storage devices, and can also be used to serve SCSI and other non-MSCP/non-TMSCP storage devices.

SCS CONNECTION: A SYSAP on one node establishes an SCS connection to its counterpart on another node. This connection will be on ONE AND ONLY ONE of the available virtual circuits.

When there are multiple virtual circuits between two OpenVMS systems it is possible for the VMS$VAXCLUSTER to VMS$VAXCLUSTER connection to use any one of these circuits. All lock traffic between the two systems will then travel on the selected virtual circuit.

Each port has a "LOAD CLASS" associated with it. This load class helps to determine which virtual circuit a connection will use. If one port has a higher load class than all others then this port will be used. If two or more ports have equally high load classes then the connection will use the

first of these that it finds. Normally all CI and DSSI ports have a load class of 14(hex), the Ethernet/FDDI port has a load class of A(hex).

For instance, if you have multiple DSSI busses and an FDDI, the VMS$VAXCLUSTER connection will chose the DSSI bus as this path has the system disk, and thus will always be the first DSSI bus discovered when the OpenVMS system boots.

To force all lock traffic off the DSSI and on to the FDDI, an adjustment to the load class value is required, or the SCS port must be disabled.

Note that with PE ports, you can typically immediately re-enable the path, permitting failover to occur should congestion or a problem arise—a running average of the path latency is checked when the virtual circuit is formed, and at periodic intervals (circa every three seconds), and when a problem with a virtual circuit arises.

In the case of PEDRIVER, the driver handles load balancing among the available Ethernet and FDDI connections based on the lowest latency path available to it. Traffic will be routed through that path until an event occurs that requires a fail-over.

In most OpenVMS versions, you can use the tools:

- SYS$EXAMPLES:LAVC$STOP_BUS

- SYS$EXAMPLES:LAVC$START_BUS

These tools permit you to disable or enable all SCS traffic on the on the specified paths.

You can also use a prefered path mechanism that tells the local MSCP disk driver (DUDRIVER) which path to a disk should be used. Generally, this is used with dual-pathed disks, forcing I/O traffic through one of the controllers instead of the other. This can be used to implement a crude form of I/O load balancing at the disk I/O level.

Prior to V7.2, the prefered path feature uses the tool:

- SYS$EXAMPLES:PREFER.MAR

In OpenVMS V7.2 and later, you can use the following DCL command:

```
$ SET PREFERED_PATH
```

The prefered path mechanism does not disable nor affect SCS operations on the non-prefered path.

15.6.2 Cluster System Parameter Settings?

The following sections contain details of configuring cluster-related system parameters.

15.6.2.1 What is the correct value for EXPECTED_VOTES in a VMScluster?

The VMScluster connection manager uses the concept of votes and quorum to prevent disk and memory data corruptions—when sufficient votes are present for quorum, then access to resources is permitted. When sufficient votes are not present, user activity will be blocked. The act of blocking user activity is called a "quorum hang", and is better thought of as a "user data integrity interlock". This mechanism is designed to prevent a partitioned VMScluster, and the resultant massive disk data corruptions.

On each OpenVMS node in a VMScluster, one sets two values in SYSGEN: VOTES, and EXPECTED_VOTES. The former is how many votes the node contributes to the VMScluster. The latter is the total number of votes expected when the full VMScluster is bootstrapped.

Some sites erroneously attempt to set EXPECTED_VOTES too low, believing that this will allow when only a subset of voting nodes are present in a VMScluster. It does not. Further, an erroneous setting in EXPECTED_VOTES is automatically corrected once VMScluster connections to other nodes are established; user data is at risk of severe corruptions during the earliest and most vulnerable portion of the system bootstrap, before the connections have been established.

One can operate a VMScluster with one, two, or many voting nodes. With any but the two-node configuration, keeping a subset of the nodes active when some nodes fail can be easily configured. With the two-node configuration, one must use a primary-secondary configuration (where the primary has all the votes), a peer configuration (where when either node is down, the other hangs), or (preferable) a shared quorum disk.

Use of a quorum disk does slow down VMScluster transitions somewhat – the addition of a third voting node that contributes the vote(s) that would be assigned to the quorum disk makes for faster transitions—but the use of a quorum disk does mean that either node in a two-node VMScluster configuration can operate when the other node is down.

If you choose to use a quoum disk, a QUORUM.DAT file will be automatically created when OpenVMS first boots and when a quorum disk is specified – well, the QUORUM.DAT file will be created when OpenVMS is booted without also needing the votes from the quorum disk.

In a two-node VMScluster with a shared storage interconnect, typically each node has one vote, and the quorum disk also has one vote. EXPECTED_VOTES is set to three.

Using a quorum disk on a non-shared interconnect is unnecessary—the use of a quorum disk does not provide any value, and the votes assigned to the quorum disk should be assigned to the OpenVMS host serving access to the disk.

For information on quorum hangs, see the OpenVMS documentation. For information on changing the EXPECTED_VOTES value on a running system, see the SET CLUSTER/EXPECTED_VOTES command, and see the documentation for the AMDS and Availability Manager tools. Also of potential interest is the OpenVMS system console documentation for the processor-specific console commands used to trigger the IPC (Interrrupt Priority Level %x0C; IPL C) handler. AMDS, Availability Manager, and the IPC handler can each be used to clear a quorum hang. Use of AMDS and Availability Manager is generally recommended over IPC, particularly because IPC can cause CLUEXIT bugchecks if the system should remain halted beyond the cluster sanity timer limits.

The quorum scheme is a set of "blade guards" deliberately implemented by OpenVMS Engineering to provide data integrity—remove these blade guards at your peril. OpenVMS Engineering did not implement the quorum mechanism to make your life more difficult—quorum was implemented to keep your data from getting scrambled.

15.6.2.2 Explain disk (or tape) allocation class settings?

The allocation class mechanism provides the system manager with a way to configure and resolve served and direct paths to storage devices within a cluster. Any served device that provides multiple paths should be configured using a non-zero allocation class, either at the MSCP (or TMSCP) storage controllers, at the port (for port allocation classes), or at the OpenVMS MSCP (or TMSCP) server. All controllers or servers providing a path to the same device should have the same allocation class (at the port, controller, or server level).

Each disk (or tape) unit number used within a non-zero disk (or tape) allocation class must be unique, regardless of the particular device prefix. For the purposes of multi-path device path determination, any disk (or tape) device with the same unit number and the same disk (or tape) allocation class configuration is assumed to be the same device.

If you are reconfiguring disk device allocation classes, you will want to avoid the use of allocation class one (1) until/unless you have Fibre Channel storage configured. (Fibre Channel storage specifically requires the use of allocation class 1. eg: 1DGA0:.)

15.6.2.2.1 How to configure allocation classes and Multi-Path SCSI?

The HSZ allocation class is applied to devices, starting with OpenVMS V7.2. It is considered a port allocation class (PAC), and all device names with a PAC have their controller letter forced to "A". (You might infer from the the text in the "Guidelines for OpenVMS Cluster Configurations" that this is something you have to do, though OpenVMS will thoughtfully handle this renaming for you.)

You can force the device names back to DKB by setting the HSZ allocation class to zero, and setting the PKB PAC to -1. This will use the host allocation class, and will leave the controller letter alone (that is, the DK controller letter will be the same as the SCSI port (PK) controller). Note that this won't work if the HSZ is configured in multibus failover mode. In this case, OpenVMS requires that you use an allocation class for the HSZ.

When your configuration gets even moderately complex, you must pay careful attention to how you assign the three kinds of allocation class: node, port and HSZ/HSJ, as otherwise you could wind up with device naming conflicts that can be painful to resolve.

The display-able path information is for SCSI multi-path, and permits the multi-path software to distinguish between different paths to the same device. If you have two paths to 1DKA100, for example by having two KZPBA controllers and two SCSI buses to the HSZ, you would have two UCBs in a multi-path set. The path information is used by the multi-path software to distinguish between these two UCBs.

The displayable path information describes the path; in this case, the SCSI port. If port is PKB, that's the path name you get. The device name is no longer completely tied to the port name; the device name now depends on the various allocation class settings of the controller, SCSI port or node.

15–14

The reason the device name's controller letter is forced to "A" when you use PACs is because a shared SCSI bus may be configured via different ports on the various nodes connected to the bus. The port may be PKB on one node, and PKC on the other. Rather obviously, you will want to have the shared devices use the same device names on all nodes. To establish this, you will assign the same PAC on each node, and OpenVMS will force the controller letter to be the same on each node. Simply choosing "A" was easier and more deterministic than negotiating the controller letter between the nodes, and also parallels the solution used for this situation when DSSI or SDI/STI storage was used.

To enable port allocation classes, see the SYSBOOT command SET/BOOT, and see the DEVICE_NAMING system parameter.

This information is also described in the Cluster Systems and Guidelines for OpenVMS Cluster Configurations manuals.

15.6.3 Tell me about SET HOST/DUP and SET HOST/HSC

The OpenVMS DCL commands SET HOST/DUP and SET HOST/HSC are used to connect to storage controllers via the Diagnostics and Utility Protocol (DUP). These commands require that the FYDRIVER device driver be connected. This device driver connection is typically performed by adding the following command(s) into the system startup command procedure:

On OpenVMS Alpha:

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> IO CONNECT FYA0/NOADAPTER/DRIVER=SYS$FYDRIVER
```

On OpenVMS VAX:

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> CONNECT FYA0/NOADAPTER
```

Alternatives to the DCL SET HOST/DUP command include the console SET HOST command available on various mid- to recent-vintage VAX consoles:

Access to Parameters on an Embedded DSSI controller:

```
SET HOST/DUP/DSSI[/BUS:{0:1}] dssi_node_number PARAMS
```

Access to Directory of tools on an Embedded DSSI controller:

```
SET HOST/DUP/DSSI[/BUS:{0:1}] dssi_node_number DIRECT
```

Access to Parameters on a KFQSA DSSI controller:

```
SHOW UQSSP ! to get port_controller_number PARAMS
SET HOST/DUP/UQSSP port_controller_number PARAMS
```

These console commands are available on most MicroVAX and VAXstation 3xxx series systems, and most (all?) VAX 4xxx series systems. For further information, see the system documentation and—on most VAX systems—see the console HELP text.

EK-410AB-MG, _DSSI VAXcluster Installation and Troubleshooting_, is a good resource for setting up a DSSI VMScluster on OpenVMS VAX nodes. (This manual predates coverage of OpenVMS Alpha systems, but gives good coverage to all hardware and software aspects of setting up a DSSI-based VMScluster—and most of the concepts covered are directly applicable to OpenVMS Alpha systems. This manual specifically covers the hardware, which is something not covered by the standard OpenVMS VMScluster documentation.)

Also see Section 15.3.3, and for the SCS name of the OpenVMS host see Section 5.6.

15.6.4 How do I rename a DSSI disk (or tape?)

If you want to renumber or rename DSSI disks or DSSI tapes, it's easy—if you know the secret incantation...

From OpenVMS:

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> CONNECT FYA0/NOADAPTER
SYSGEN> ^Z
$ SET HOST/DUP/SERV=MSCP$DUP/TASK=PARAMS <DSSI-NODE-NAME>
...
PARAMS> STAT CONF
<The software version is normally near the top of the display.>
PARAMS> EXIT
...
```

From the console on most 3000- and 4000-class VAX system consoles... (Obviously, the system must be halted for these commands...)

Integrated DSSI:

```
SET HOST/DUP/DSSI[/BUS:[0:1]] dssi_node_number PARAMS
```

KFQSA:

```
SET HOST/DUP/UQSSP port_controller_number PARAMS
```

15–16

For information on how to get out into the PARAMS subsystem, also see the HELP at the console prompt for the SET HOST syntax, or see the HELP on SET HOST /DUP (once you've connected FYDRIVER under OpenVMS).

Once you are out into the PARAMS subsystem, you can use the FORCEUNI option to force the use of the UNITNUM value and then set a unique UNITNUM inside each DSSI ISE—this causes each DSSI ISE to use the specfied unit number and not use the DSSI node as the unit number. Other parameters of interest are NODENAME and ALLCLASS, the node name and the (disk or tape) cluster allocation class.

Ensure that all disk unit numbers used within an OpenVMS Cluster disk allocation class are unique, and all tape unit numbers used within an OpenVMS Cluster tape allocation class are also unique. For details on the SCS name of the OpenVMS host, see Section 5.6. For details of SET HOST/DUP, see Section 15.6.3.

15.6.5 Where can I get Fibre Channel Storage (SAN) information?

- http://www.openvms.compaq.com/openvms/fibre/index.html

15.6.6 How can I split up an OpenVMS Cluster?

Review the VMScluster documentation, and the System Management documentation. The following are the key points, but are likely not the only things you will need to change.

OpenVMS Cluster support is directly integrated into the operating system, and there is no way to remove it. You can, however, remote site-specific tailoring that was added for a particular cluster configuration.

First: Create restorable image BACKUPs of each of the current system disks. If something gets messed up, you want a way to recover, right?

Create standalone BACKUP kits for the OpenVMS VAX systems, and create or acquire bootable BACKUP kits for the OpenVMS Alpha systems.

Use CLUSTER_CONFIG or CLUSTER_CONFIG_LAN to remove the various system roots and to shut off boot services and VMScluster settings.

Create as many architecture-specific copies of the system disks as required. Realize that the new systems will all likely be booting through root SYS0—if you have any system-specific files in any other roots, save them.

Relocate the copies of the VMScluster common files onto each of the new system disks.

Reset the console parameters and boot flags on each system for use on a standalone node.

Reset the VAXCLUSTER and NISCS_LOAD_PEA0 parameters to 0 in SYSGEN and in MODPARAMS.DAT.

Clobber the VMScluster group ID and password using SYSMAN.

Reboot the systems seperately, and run AUTOGEN on each.

Shut off MOP services via NCP or LANCP on the boot server nodes.

Permanent seperation also requires the duplication of shared files. For a list of the files commonly shared, please see the most current version of SYS$STARTUP:SYLOGICALS.TEMPLATE, and specifically a version from OpenVMS V7.2 or later. The following files are typically shared within a cluster:

```
Filename:                default directory (in common root) and file type:
  SYSUAF                      SYS$SYSTEM:.DAT
  SYSUAFALT                   SYS$SYSTEM:.DAT
  SYSALF                      SYS$SYSTEM:.DAT
  RIGHTSLIST                  SYS$SYSTEM:.DAT
  NETPROXY                    SYS$SYSTEM:.DAT
  NET$PROXY                   SYS$SYSTEM:.DAT
  NETOBJECT                   SYS$SYSTEM:.DAT
  NETNODE_REMOTE              SYS$SYSTEM:.DAT
  QMAN$MASTER                 SYS$SYSTEM: (this is a set of related files)
  LMF$LICENSE                 SYS$SYSTEM:.LDB
  VMSMAIL_PROFILE             SYS$SYSTEM:.DATA
  VMS$OBJECTS                 SYS$SYSTEM:.DAT
  VMS$AUDIT_SERVER            SYS$MANAGER:.DAT
  VMS$PASSWORD_HISTORY        SYS$SYSTEM:.DATA
  NETNODE_UPDATE              SYS$MANAGER:.COM
  VMS$PASSWORD_POLICY         SYS$LIBRARY:.EXE
  LAN$NODE_DATABASE           SYS$SYSTEM:LAN$NODE_DATABASE.DAT
```

Also see the topics on "cluster divorce" in the Ask The Wizard area.

- http://www.openvms.compaq.com/wizard/

- http://www.openvms.compaq.com/wizard/wizard.zip

For additional information, please see Section 3.9.

Information on changing node names is included in Section 5.6.

15.6.7 Details on Volume Shadowing?

This section contains information on host-based volume shadowing; on the disk mirroring capabilities available within OpenVMS.

15.6.7.1 Does volume shadowing require a non-zero allocation classes?

Yes, use of host-based Volume Shadowing requires that the disk(s) involved be configured in a non-zero allocation class.

Edit SYS$SYSTEM:MODPARAMS.DAT to include a declaration of an non-zero allocation class, such as setting the host allocation class to the value 7:

```
ALLOCLASS = 7
```

Then AUTOGEN the system, and reboot.

You should now be able to form the shadow set via a command such as the following:

```
$ MOUNT dsa1007: /SHADOW=($7$dkb300:,$7$dkb500:) volumelabel
```

When operating in an OpenVMS Cluster, this sequence will typically change the disk names from the SCSNODE prefix (scsnode$dkann) to the allocation-class prefix (7dkannn). This may provide you with the opportunity to move to a device-independent scheme using logical name constructs such as the DISK$volumelabel logical names in your startup and application environments; an opportunity to weed out physical device references.

Allocation class one is used by Fibre Channel devices; it can be best to use another non-zero allocation class even if Fibre Channel is not currently configured and not currently planned.

15.6.7.2 Volume Shadowing MiniCopy vs MiniMerge?

MiniMerge support has been available for many years with OpenVMS host-based volume shadowing, so long as you had MSCP controllers (eg: HSC, HSJ, or HSD) which supported the Volume Shadowing Assist called "Write History Logging".

If you want minimerges on HSG80 (Fibre Channel) controllers, please see the "Fibre Channel in a Disaster-Tolerant OpenVMS Cluster System" whitepaper at:

- http://www.openvms.compaq.com/openvms/fibre/fc_hbvs_dtc_wp.pdf

Minimerge support on HSG80 is expected to require ACS 8.7 and OpenVMS Alpha V7.3-1, assuming the development goes according to plan.

The following sections describe both MiniCopy and MiniMerge, and can provide a basis for discussions.

15.6.7.2.1   MiniCopy?

A Shadowing Full Copy occurs when you add a disk to an existing shadowset using a MOUNT command; the entire contents of the disk are effectively copied to the new member (using an algorithm that goes through in 127-block increments and reads one member, compares with the target disk, and if the data differs, writes the data to the target disk and loops back to the read step, until the data is equal for that 127-block section). (This is one of the reasons why the traditional recommendation for adding new volumes to a shadowset was to use a BACKUP/PHYSICAL copy of an existing shadowset volume, simply because the reads then usually matched and thus shadowing usually avoided the need for the writes.)

If you warn OpenVMS ahead of time (at dismount time) that you're planning to remove a disk from a shadowset but re-add it later, OpenVMS will keep a bitmap tracking what areas of the disk have been modified while the disk was out of the shadowset, and when you re-add it later with a MOUNT command OpenVMS only has to update the areas of the returned disk that the bit-map indicates are now out-of-date. OpenVMS does this with a read source / write target algorithm, which is much faster than the shenanigans the Full Copy does, so even if all of the disk has changed, a MiniCopy is faster than a Full Copy.

15.6.7.2.2 MiniMerge?

A Shadowing Merge is initiated when an OpenVMS node in the cluster (which had a shadowset mounted) crashes or otherwise leaves unexpectedly, without dismounting the shadowset first. In this case, OpenVMS must ensure that the data is identical, since Shadowing guarantees that the data on the disks in a shadowset will be identical. In a regular Merge operation, Shadowing uses an algorithm similar to the Full Copy algorithm (except that it can choose either of the members' contents as the source data, since both are considered equally valid), and scans the entire disk. Also, to make things worse, for any read operations in the area ahead of what has been merged, Shadowing will first merge the area containing the read data, then allow the read to occur.

A Merge can be very time-consuming and very I/O intensive, so some controllers have Shadowing Assists to make it faster. If the controllers support Write History Logging, the controllers record the areas (LBNs) that are the subject of Shadowing writes, and if a node crashes, the surviving nodes can query the controllers to find out what exact areas of the disk the departed node was writing to just before the crash, and thus Shadowing only needs to merge just those few areas, so this tends to take seconds as opposed to hours for a regular Merge.

Index

1858
    17-Nov-1858 • 4–2
2000
    Year • 5–26

A

ACCVIO • 5–26
ACTS • 4–7
Allocation Class
    Fibre Channel • 15–19
    Non-Zero • 15–19
    Volume Shadowing • 15–19
ALLOCLASS system parameter • 15–19
ANSI C • 10–14
AP • 10–4
Argument Pointer • 10–4
Ask The Wizard • 3–10, 5–17, 5–18, 5–25, 8–5,
    9–2, 10–2, 10–5, 10–6, 10–12, 10–17,
    10–19, 15–1, 15–2, 15–19
ATW • 3–10, 5–17, 5–18, 5–25, 8–5, 9–2, 10–2,
    10–5, 10–6, 10–12, 10–17, 10–19, 15–1,
    15–2, 15–19
AUTOGEN • 5–8
Automatic Foreign Command • 8–1
AVI • 7–2

B

Backdrops • 11–7
Bad Block Handling • 9–2
BB_WATCH • 4–5
BC16E • 14–36
BCKTRNSFAIL • 15–4
Bi-Endian • 10–13
Big-Endian • 10–13
BIND • 15–4
BNU Bookreader • 3–1
Bookreader
    BNU • 3–1
    MGBOOK • 3–1

C

C • 10–14, 10–20
    Compaq C Compiler • 10–14, 10–20
    DEC C Compiler • 10–14, 10–20
    HP C Compiler • 10–14, 10–20
    VAX C Compiler • 10–14
Caesar
    Julius • 4–2
Carrier Sense Multiple Access, with Collision
    Detect • 14–38
CD
    Recording • 9–6, 14–32
CD-Audio • 7–1
CD Player • 7–1
CD-R • 9–6, 14–32
CDRECORD • 9–6, 14–32
CD-RW • 9–6, 14–32
CDWRITE • 9–6
cli$get_value • 10–2
Clock
    TODR • 4–4, 4–6
    TOY • 4–4, 4–6
Clockmeister, Ulysses T.
    See Time
CMA$RTL.EXE • 10–20
$cmsupr • 12–4
COMMON • 10–4
Console
    VAXstation 3100 • 9–3
$creprc • 10–6
CSLG • 2–6
CSMA/CD • 14–38

D

DATMISCH • 10–14
Daylight Savings Time • 4–10, 10–17

T