VMS Version 5.4 Upgrade and Installation Manual

Order Number: AA–NG61C–TE

June 1990

This document contains step-by-step instructions for upgrading and installing Version 5.4 of the VMS operating system and VMS DECwindows software.

Revision/Update Information: This manual supersedes the *VMS Version 5.3 Upgrade and Installation Procedures*.

Software Version: VMS Version 5.4

digital equipment corporation
maynard, massachusetts

June 1990

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

Restricted Rights: Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013.

© Digital Equipment Corporation 1990.

All Rights Reserved.
Printed in U.S.A.

The postpaid Reader's Comments forms at the end of this document request your critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

CDA	DEQNA	MicroVAX	VAX RMS
DDIF	Desktop–VMS	PrintServer 40	VAXserver
DEC	DIGITAL	Q-bus	VAXstation
DECdtm	GIGI	ReGIS	VMS
DECnet	HSC	ULTRIX	VT
DECUS	LiveLink	UNIBUS	XUI
DECwindows	LN03	VAX	
DECwriter	MASSBUS	VAXcluster	digital™

The following are third-party trademarks:

Postscript and Display PostScript are registered trademarks of Adobe Systems Incorporated.

ZK5487

Production Note

This book was produced with the VAX DOCUMENT electronic publishing system, a software tool developed and sold by Digital. In this system, writers use an ASCII text editor to create source files containing text and English-like code; this code labels the structural elements of the document, such as chapters, paragraphs, and tables. The VAX DOCUMENT software, which runs on the VMS operating system, interprets the code to format the text, generate a table of contents and index, and paginate the entire document. Writers can print the document on the terminal or line printer, or they can use Digital-supported devices, such as the LN03 laser printer and PostScript printers (PrintServer 40 or LN03R ScriptPrinter), to produce a typeset-quality copy containing integrated graphics.

Contents

PREFACE xv

PART I

CHAPTER 1 THE INSTALLATION AND UPGRADE PROCEDURES 1–1

1.1 WHAT HAPPENS DURING AN INSTALLATION 1–1

1.2 WHAT HAPPENS DURING AN UPGRADE 1–3

PART II

CHAPTER 2 BEFORE INSTALLING VMS 2–1

2.1 PRE-INSTALLATION CHECKLIST 2–1

2.2 THE DISTRIBUTION MEDIA AND THE SYSTEM DISK 2–2

2.3 LOCAL DRIVES 2–2

2.4 HSC DRIVES 2–2

2.5 PREPARING FOR A VMS INSTALLATION 2–3
2.5.1 Preparing for a VAXcluster Environment 2–3
2.5.2 Preparing Disk and Tape Drives 2–4

Contents

2.6 INFORMATION YOU NEED DURING THE INSTALLATION 2–5

CHAPTER 3 INSTALLING THE VMS OPERATING SYSTEM AND DECWINDOWS SOFTWARE 3–1

3.1 CAUTIONS AND RESTRICTIONS 3–1

3.2 BOOTING STANDALONE BACKUP 3–2

3.3 CREATING THE SYSTEM DISK 3–3

3.4 INSTALLING VMS DECWINDOWS SOFTWARE 3–8

3.5 JOINING A VAXCLUSTER 3–9

3.6 SETTING PASSWORDS 3–12

3.7 APPLYING THE MANDATORY UPDATE 3–14

3.8 REGISTERING LICENSES 3–15

3.9 RUNNING AUTOGEN 3–16

CHAPTER 4 AFTER INSTALLING VMS 4–1

4.1 POST-INSTALLATION CHECKLIST 4–1

4.2 REGISTERING YOUR LICENSES 4–2

4.3 BACKING UP YOUR SYSTEM DISK 4–2

4.4 CUSTOMIZING THE SYSTEM 4–2

4.5 CONFIGURING AND STARTING DECNET–VAX 4–3

4.6 CUSTOMIZING OR DISABLING DECDTM SERVICES 4–3

4.7 TESTING THE SYSTEM 4–4

4.8 DECOMPRESSING THE SYSTEM LIBRARIES 4–4

4.9 SETTING UP THE SYSTEM MESSAGES HELP LIBRARY 4–4

4.10 REMOVING UNWANTED FILES 4–5

4.11 CUSTOMIZING DECWINDOWS 4–6

4.12 BACKING UP THE CUSTOMIZED SYSTEM DISK 4–6

4.13 RUNNING AUTOGEN 4–6

PART III

CHAPTER 5 OVERVIEW OF THE UPGRADE PROCEDURE 5–1

5.1 WHAT HAPPENS DURING AN UPGRADE 5–2

5.2 CAUTIONS AND RESTRICTIONS 5–2

5.3 ITEMS YOU NEED FOR THE UPGRADE 5–3

CHAPTER 6 BEFORE YOU DO AN UPGRADE 6–1

6.1 PRE-UPGRADE CHECKLIST 6–1

6.2 UNDERSTANDING VERSION 5.4 OF THE VMS OPERATING SYSTEM 6–2

6.3 CHECKING AUTOGEN FEEDBACK DATA 6–2

6.4 BACKUP TASKS 6–3

6.5 PREPARING THE SYSTEM DISK 6–4

6.6 PREPARING THE SYSTEM 6–7

6.7 INFORMATION YOU NEED DURING THE UPGRADE 6–8

CHAPTER 7 UPGRADING A VAXCLUSTER ENVIRONMENT 7–1

7.1 PERFORMING A CONCURRENT UPGRADE 7–1

7.2 ROLLING UPGRADE OVERVIEW 7–2

7.2.1 Performing a Rolling Upgrade 7–3

7.2.1.1 Special Steps Required for Version 5.3 Systems • 7–3

7.2.1.2 Rolling Upgrade Instructions • 7–3

CHAPTER 8 STANDALONE UPGRADE 8–1

8.1 PREPARING TO BOOT FROM [SYSF] 8–1

8.2 CONFIGURING DEVICES 8–2

8.3 BEGINNING THE UPGRADE PROCEDURE 8–2

8.4 UPGRADE PHASE 1 8–5

8.5 UPGRADE PHASE 2 8–10

8.6 UPGRADE PHASE 3 8–11

8.7 UPGRADE PHASE 4 8–12

8.8 UPGRADE PHASE 5 8–13

8.9 UPGRADE PHASE 6 8–13

CHAPTER 9 AFTER YOU DO AN UPGRADE 9–1

9.1 POST-UPGRADE CHECKLIST 9–1

9.2 REPLACING SYSUAF.DAT 9–2

9.3 EXAMINING AGEN$PARAMS.REPORT 9–2

9.4 CHECKING THE SIZE OF SYSTEM FILES 9–2

9.5 EXAMINING YOUR COMMAND PROCEDURES 9–3

9.6 CUSTOMIZING OR DISABLING DECDTM SERVICES 9–3

9.7 CHANGING MODPARAMS.DAT 9–4

9.8 UPDATING YOUR DECNET–VAX CONFIGURATION 9–4
9.8.1 Running NETCONFIG_UPDATE.COM 9–4
9.8.2 Running UPDATE_CLUSTER_MEMBERS.COM 9–7

9.9	UPDATING YOUR CONSOLE MEDIA	9–7
9.9.1	VAX 8530, 8550, 8810 (8700), 8820-N (8800)	9–8
9.9.2	VAX 11/730, 750, 780, 785 and VAX 8200, 8250, 8300, 8350, 8600, 8650	9–9
9.9.3	VAX 9000	9–10
9.10	DECOMPRESSING THE SYSTEM LIBRARIES	9–10
9.11	SETTING UP THE SYSTEM MESSAGES HELP LIBRARY	9–10
9.12	TESTING THE SYSTEM WITH UETP	9–11
9.13	INCREASING FREE SPACE ON THE SYSTEM DISK	9–12
9.13.1	Using VMSTAILOR and DECW$TAILOR	9–12
9.13.2	Purging System Files	9–12
9.14	CUSTOMIZING VMS DECWINDOWS SOFTWARE	9–12
9.15	BACKING UP THE SYSTEM DISK	9–13
9.16	RUNNING AUTOGEN	9–13

PART IV

CHAPTER 10	RUNNING UETP	10–1
10.1	SUMMARY OF UETP OPERATING INSTRUCTIONS	10–1
10.2	LOGGING IN	10–3
10.2.1	SYSTEST Directories	10–3
10.3	SETTING UP FOR UETP	10–3
10.3.1	The System Disk	10–4
10.3.2	Additional Disks	10–4
10.3.3	Magnetic Tape Drives	10–5

10.3.4 Tape Cartridge Drives — 10–6
10.3.4.1 TLZ04 Tape Drives • **10–6**
10.3.5 Compact Disc Drives — 10–6
10.3.6 Optical Disk Drives — 10–6
10.3.7 Terminals and Line Printers — 10–7
10.3.8 Ethernet Adapters — 10–7
10.3.9 The DR11–W — 10–7
10.3.10 The DRV11–WA — 10–8
10.3.11 The DR750 or DR780 — 10–8
10.3.12 The MA780 — 10–8
10.3.13 A Second LPA11–K — 10–9
10.3.14 Devices Not Tested — 10–9
10.3.15 VAXcluster Testing — 10–9
10.3.16 A Small-Disk System — 10–10
10.3.17 DECnet–VAX — 10–10

10.4 PREPARING VECTOR PROCESSORS AND THE VAX VECTOR INSTRUCTION EMULATION FACILITY (VVIEF) — 10–11

10.5 STARTING UETP — 10–11
10.5.1 Running a Subset of Phases — 10–12
10.5.2 Single Run Versus Multiple Passes — 10–12
10.5.3 Defining User Load for Load Test — 10–13
10.5.4 Report Formats — 10–13
10.5.5 Termination of UETP — 10–14
10.5.5.1 Using CTRL/Y • **10–14**
10.5.5.2 Using CTRL/C • **10–15**

10.6 TROUBLESHOOTING — 10–15
10.6.1 Relationship of UETP to Error Logging and Diagnostics — 10–15
10.6.2 Interpreting UETP Output — 10–16
10.6.2.1 Defining a Remote Node for UETP Ethernet Testing • **10–17**
10.6.3 Log Files — 10–18
10.6.4 Possible UETP Errors — 10–19
10.6.4.1 Wrong Quotas, Privileges, or Account • **10–20**
10.6.4.2 UETINIT01 Failure • **10–21**
10.6.4.3 Device Allocated or in Use by Another Application • **10–23**
10.6.4.4 Insufficient Disk Space • **10–23**
10.6.4.5 Incorrect Setup of a VAXcluster • **10–24**
10.6.4.6 Problems During the Load Test • **10–25**
10.6.4.7 DECnet–VAX Error • **10–26**
10.6.4.8 Errors Logged but Not Displayed • **10–27**
10.6.4.9 No PCB or Swap Slots • **10–27**
10.6.4.10 Hangs • **10–28**

10.6.4.11 Lack of Default Access for the FAL Object • **10–28**
10.6.4.12 Bugchecks and Machine Checks • **10–29**

10.7 UETP TESTS AND PHASES 10–29
10.7.1 Initialization Phase 10–29
10.7.2 Device Test Phase 10–30
10.7.2.1 How the Device Phase Works • **10–30**
10.7.2.2 Running a Single Device Test • **10–30**
10.7.3 System Load Test Phase 10–33
10.7.4 DECnet–VAX Test Phase 10–34
10.7.4.1 Environment • **10–34**
10.7.4.2 How the DECnet–VAX Phase Works • **10–35**
10.7.5 Cluster-Integration Test Phase 10–37

CHAPTER 11 CUSTOMIZING VMS DECWINDOWS SOFTWARE 11–1

11.1 CONTROLLING STARTUP OF VMS DECWINDOWS SOFTWARE 11–1
11.1.1 After an Installation 11–1
11.1.2 After an Upgrade 11–1
11.1.3 VMS DECwindows Logical Names Created At Startup 11–2

11.2 CUSTOMIZING THE SERVER STARTUP 11–3

11.3 USING TCP/IP WITH DECWINDOWS 11–7

11.4 USING CUSTOMER-WRITTEN TRANSPORTS WITH DECWINDOWS 11–7

11.5 USING THE EXAMPLE TRANSPORT 11–8

11.6 DEFINING PRINT FORMATS 11–9

11.7 COMPILING ADA SOURCE FILES 11–10

CHAPTER 12 USING VMSTAILOR AND DECW$TAILOR 12–1

12.1 USING VMSTAILOR 12–1
12.1.1 Removing VMS Files 12–3
12.1.2 Adding VMS Files 12–4

12.2 USING DECW$TAILOR 12–7
12.2.1 Removing DECwindows Files 12–8
12.2.2 Adding DECwindows Files 12–9

APPENDIX A BOOTING FROM [SYSF] DURING AN UPGRADE A–1

A.1 MICROVAX AND VAXSTATION COMPUTERS A–1

A.2 VAX 8530, 8550, 8810 (8700), 8820-N (8800), 8820, 8830, AND 8840 A–1

A.3 VAX 11/750, VAX 8200, 8250, 8300, AND 8350 A–2

A.4 VAX 11/730, VAX 11/780, VAX 11/785, VAX 8600, AND VAX 8650 A–3

A.5 VAX 6000 SERIES A–5

A.6 VAX 9000 SERIES A–5

APPENDIX B FILES IN THE VMS LIBRARY AND OPTIONAL SAVE SETS B–1

APPENDIX C DECWINDOWS SOFTWARE COMPONENTS TABLE C–1

APPENDIX D LAYERED PRODUCTS CAUTION D–1

Contents

GLOSSARY Glossary–1

INDEX

EXAMPLES

12–1 Sample VMSTAILOR Session 12–6
12–2 Sample DECW$TAILOR Session 12–10

FIGURES

1–1 Documentation Used to Perform a VMS Installation 1–2
1–2 Documentation Used to Perform a VMS Upgrade 1–4

TABLES

3–1 Installation Questions for CI-Only VAXclusters 3–10
3–2 Installation Questions for Local Area and Mixed-Interconnect VAXclusters 3–11
10–1 Device Tests 10–33
11–1 DECW$LOGICAL_NAMES 11–2
11–2 DECwindows Keymap 11–4
11–3 DECW$SERVER(n)_TABLE 11–6
11–4 Print Dialog Box Logical Names 11–9
12–1 Classes of VMS Files That Can Be Added or Removed with VMSTAILOR 12–1
12–2 Classes of VMS Files That Can Be Added or Removed with DECW$TAILOR 12–7
C–1 DECwindows Software Components C–1
D–1 Layered Products Currently Available for VMS Version 5.4 D–1

Preface

This manual is your *primary* source of information for upgrading or installing the VMS operating system. It contains step-by-step instructions for upgrading and installing Version 5.4 of the VMS operating system and VMS DECwindows.

IMPORTANT: When you are ready to upgrade or install the VMS operating system, use the supplied documentation as follows:

1. **Read all release-specific cover letters (if any) included with your distribution kit.**
2. **If you are performing an upgrade or installation, follow the step-by-step upgrade or installation instructions in this guide.**
3. **When the instructions in this guide direct you to do so, refer to the installation and operations supplement for information specific to your VAX processor.**
4. **Store the hardcopy of this guide and the installation and operations supplement for your VAX processor in the binder that contains the *VMS Release Notes*.**

Intended Audience

This manual is intended for anyone responsible for installing or upgrading the VMS operating system. Read this manual before you install or upgrade to Version 5.4 of the VMS operating system.

Document Structure

This manual is divided into four parts.

Part I

Part I provides an overview of the upgrade and installation procedures.

- Chapter 1 describes what happens during an installation or an upgrade and refers you to the appropriate documentation.

Part II

Part II provides step-by-step instructions for installing the VMS operating system and VMS DECwindows.

- Chapter 2 describes the tasks you must perform before installing the VMS operating system.
- Chapter 3 describes how to install Version 5.4 of the VMS operating system and VMS DECwindows on a VAXcluster or a standalone VAX computer.

- Chapter 4 lists the tasks you should perform after you install the VMS operating system.

Part III

Part III provides step-by-step instructions for upgrading the VMS operating system.

- Chapter 5 contains cautions, restrictions, and requirements for upgrading your VAX computer system.
- Chapter 6 tells you how to prepare your system for an upgrade.
- Chapter 7 contains additional information for upgrading a VAXcluster environment.
- Chapter 8 describes the procedure for upgrading a single computer system.
- Chapter 9 lists the tasks you should perform after the upgrade.

Part IV

Part IV contains information on additional tasks you might perform after upgrading or installing the VMS operating system.

- Chapter 10 describes the VMS User Environment Test Package (UETP) and how to use it to test the system.
- Chapter 11 describes how to customize VMS DECwindows software.
- Chapter 12 contains information on how to add or remove optional VMS and VMS DECwindows software using VMSTAILOR and DECW$TAILOR.

This manual also contains four appendixes and a glossary.

- Appendix A describes how to boot from SYSF on the different types of VAX computers.
- Appendix B lists the files in the VMS *library* and *optional* save sets.
- Appendix C lists VMS DECwindows files.
- Appendix D outlines layered product support for Version 5.4 of the VMS operating system.
- The Glossary defines terms.

Associated Documents

The following documents might be useful:

- The VMS upgrade and installation supplement for your VAX computer—Provides supplemental information that you will need during an installation or upgrade, such as booting the system and determining device names.
- *VMS Version 5.4 New Features Manual*—Describes new software features of VMS Version 5.4.

- *VMS VAXcluster Manual*—Describes the procedures for setting up and managing VAXcluster configurations.
- *VMS Version 5.4 Release Notes*—Provides notes on the various aspects of the VMS operating system.
- *VMS License Management Utility Manual*—Provides information about registering and managing licenses on a VMS system.
- The hardware manuals supplied with your VAX computer provide detailed information about system hardware.

Conventions

The following conventions are used in this manual:

Ctrl/x	A sequence such as Ctrl/x indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.
Return	In examples, a key name is shown enclosed in a box to indicate that you press a key on the keyboard. (In text, a key name is not enclosed in a box.)
. . .	In examples, a horizontal ellipsis indicates one of the following possibilities: • Additional optional arguments in a statement have been omitted. • The preceding item or items can be repeated one or more times. • Additional parameters, values, or other information can be entered.
. . .	A vertical ellipsis indicates the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.
[ ]	In format descriptions, brackets indicate that whatever is enclosed within the brackets is optional; you can select none, one, or all of the choices. (Brackets are not, however, optional in the syntax of a directory name in a file specification or in the syntax of a substring specification in an assignment statement.)
red ink	Red ink indicates information that you must enter from the keyboard or a screen object that you must choose or click on. For online versions of the book, user input is shown in **bold**.
boldface text	Boldface text represents the introduction of a new term or the name of an argument, an attribute, or a reason. Boldface text is also used to show user input in online versions of the book.

UPPERCASE TEXT	Uppercase letters indicate that you must enter a command (for example, enter OPEN/READ), or they indicate the name of a routine, the name of a file, the name of a file protection code, or the abbreviation for a system privilege.
-	Hyphens in coding examples indicate that additional arguments to the request are provided on the line that follows.
numbers	Unless otherwise noted, all numbers in the text are assumed to be decimal. Nondecimal radixes—binary, octal, or hexadecimal—are explicitly indicated.

This manual often refers to products by their abbreviated names. For example, the VAX 8200, VAX 8250, VAX 8300, and VAX 8350 computers are referred to collectively as the VAX 8200, 8250, 8300, 8350. The MicroVAX 2000 computer is referred to as the MicroVAX 2000, and so on.

Part I

Part I contains an overview of the VMS installation and upgrade procedures.

1 The Installation and Upgrade Procedures

This chapter describes what happens during the installation and upgrade procedures. It also tells whether you should do an installation or an upgrade and refers you to the appropriate documentation.

Caution: You cannot install or upgrade to Version 5.4 of the VMS operating system on the following VAX computers and system disks:

- **MicroVAX I, VAXstation I, VAXstation 8000, and VAX 11/725 computers**
- **RK07, RL02, RC25, RD32, RD51, and RD52 system disks**

1.1 What Happens During an Installation

When you install the VMS operating system, the installation procedure does the following:

- Initializes the system disk, erasing its contents
- Creates a system directory structure
- Transfers the VMS files from the distribution media to the system disk

Use the installation procedure under the following conditions:

- If your VAX computer is new (it has never had any version of the operating system running on it).
- If your VAX computer is running a version of the VMS operating system and you want to erase the entire contents of the system disk (both VMS and user files).
- If you are running the VMS operating system, but cannot upgrade. For example, if you changed the names of system directories on the system disk, the upgrade procedure will not work correctly. You can either restore the system disk to its original directory structure, or you can re-install the VMS operating system.

Caution: The installation procedure initializes the system disk, erasing its contents. For this reason, use the installation procedure only on new VAX computers or if you want to overwrite the contents of the system disk.

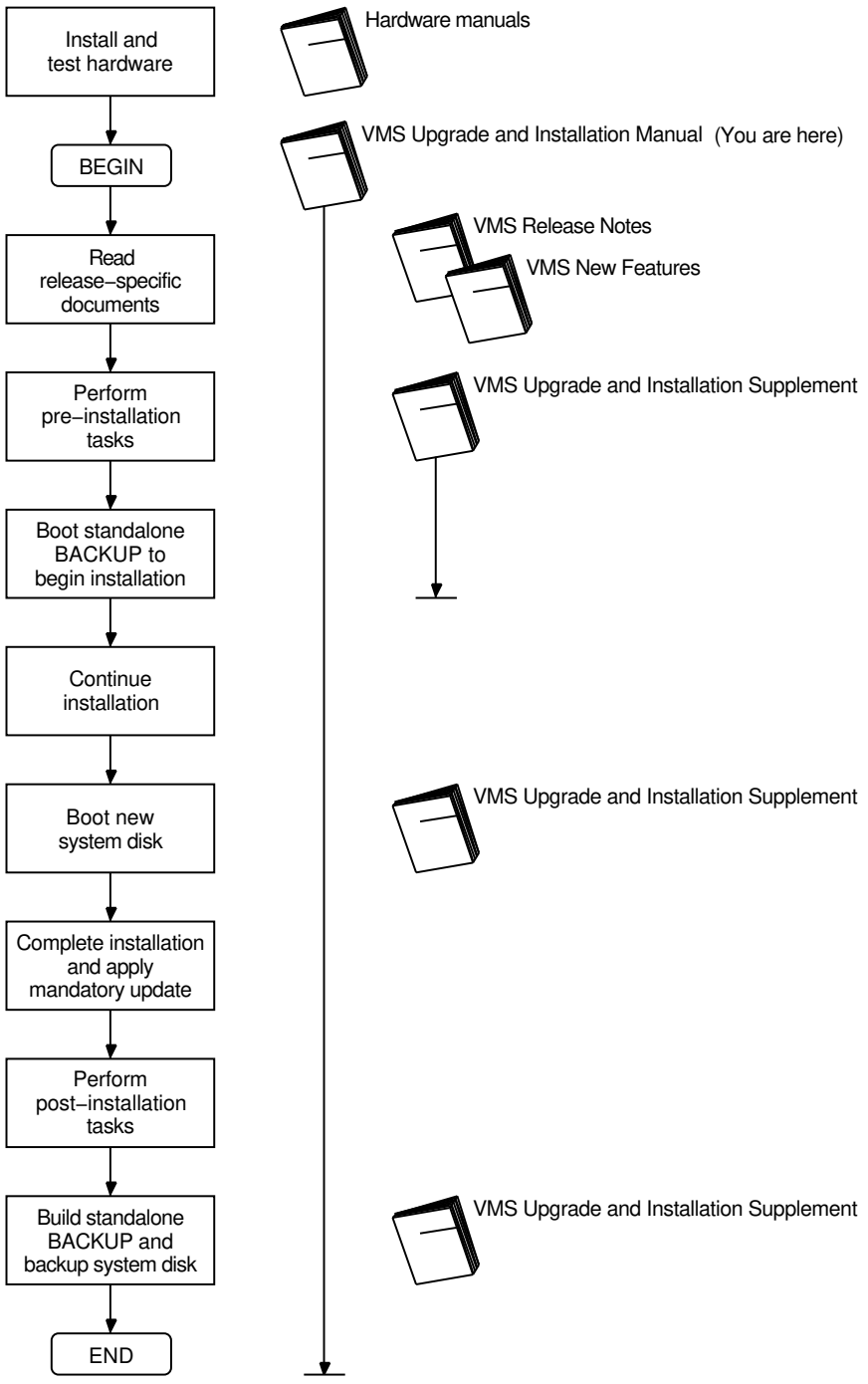
This manual contains step-by-step instructions for performing a VMS installation. However, you will need to refer to several different manuals before, and during, an installation. Figure 1–1 illustrates the major steps in an installation and the documents you will use to perform each step.

The Installation and Upgrade Procedures

1.1 What Happens During an Installation

If you are installing the VMS operating system, go to Chapter 2.

Figure 1–1 Documentation Used to Perform a VMS Installation



1.2 What Happens During an Upgrade

When you upgrade the VMS operating system, the upgrade procedure does the following:

- Makes room for the upgrade by purging and deleting some VMS files but leaves all the user files intact
- Transfers the VMS files from the distribution media to the system disk
- Merges the old VMS files and the new VMS files
- Cleans up files and structures used only during the upgrade

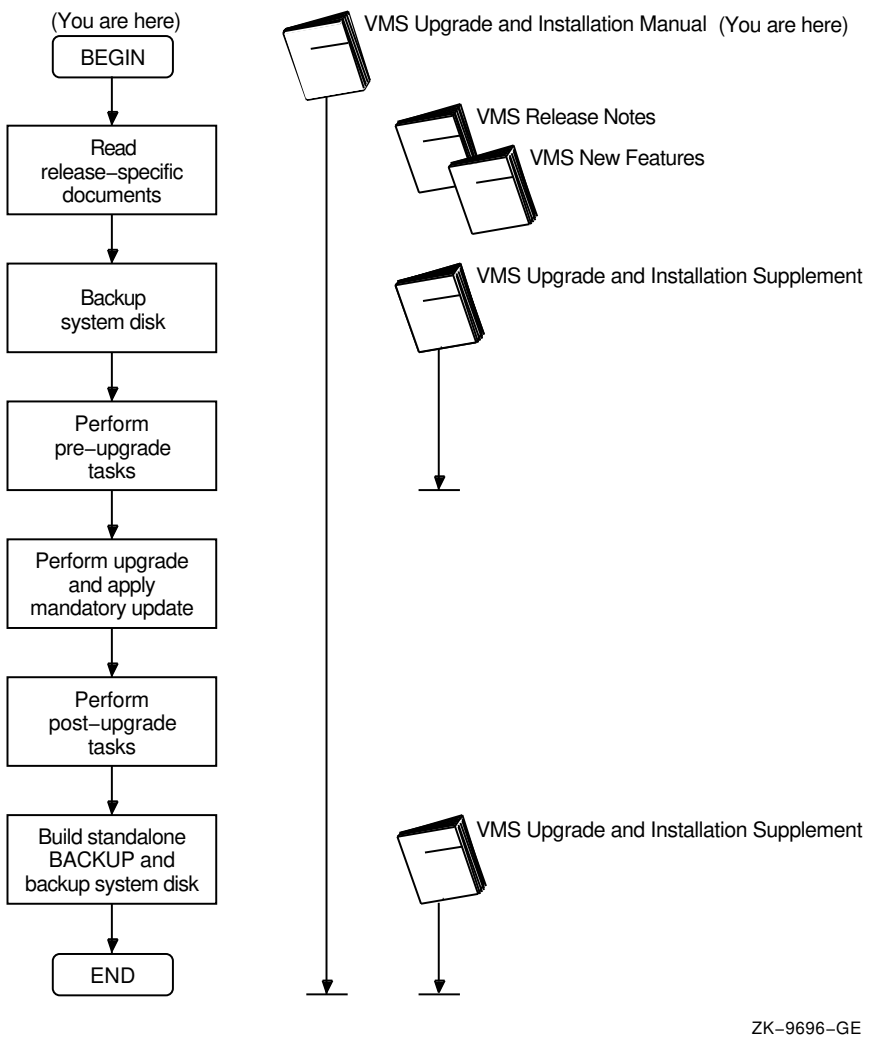
In most cases, you can use the upgrade procedure to obtain a higher version. The upgrade procedure does not initialize the system disk.

Caution: The upgrade procedure will not work correctly if you have changed the names of system directories on your system disk. Restore your VMS system disk to a standard directory structure before attempting an upgrade.

This manual contains step-by-step instructions for performing a VMS upgrade. However, you will need to refer to several different manuals before, and during, an installation. Figure 1–2 illustrates the major steps in an upgrade and the documents you will use to perform each step.

If you are upgrading your system, go to Chapter 5.

Figure 1–2 Documentation Used to Perform a VMS Upgrade



Part II

Part II describes how to install Version 5.4 of the VMS operating system and VMS DECwindows software on a VAXcluster or a single VAX computer.

2 Before Installing VMS

This chapter describes how to prepare for a VMS installation. It also provides information you must know before installing the VMS operating system. This information includes the following:

- Definition of the terms **distribution media** and **system disk**
- Description of local and HSC (hierarchical storage controller) drives

2.1 Pre-Installation Checklist

Experienced system managers can use the following checklist to make sure they have completed all the necessary tasks before beginning the installation. System Managers with less experience are advised to read this chapter beginning with Section 2.2, and to use the following checklist after completing the tasks in this chapter.

☐ Make sure the hardware is installed and tested (described in the hardware manuals for your VAX computer).

☐ Check the contents of your kit against the bill of materials.

☐ Read the *VMS Version 5.4 New Features Manual* and the *VMS Version 5.4 Release Notes* and any cover letters included with your kit.

☐ Turn on the system (described in the VMS upgrade and installation supplement for your VAX computer).

☐ Set up your system to record the installation procedure (described in the VMS upgrade and installation supplement for your computer).

☐ Read the ‘‘Before Installing VMS’’ section in the VMS upgrade and installation supplement for your computer and perform the tasks described there.

☐ If you have a VAX computer that uses console media, make sure you have the most recent version of the console media.

☐ If you are installing the VMS operating system on a VAX computer in a VAXcluster environment, determine whether you want a CI-only, local area, or mixed-interconnect cluster (described in the *VMS VAXcluster Manual*). Obtain the following information, depending on the type of VAXcluster:

— CI-Only VAXcluster: Get the allocation class value, DECnet node name, and DECnet node address for the computer. You also need to decide if you want a quorum disk, and the location of page and swap files for the system you are installing.

— Local Area and Mixed-Interconnect VAXclusters: Get the allocation class value, the DECnet node name, the DECnet node address, the cluster group number, and the cluster password. You also need to decide if you want a quorum disk, and a location for the page and swap files for the system you are installing.

☐ Prepare the disk and tape drives (described in Section 2.5.2).

☐ Make sure you have the following information (described in Section 2.6) readily available:

- Passwords of at least 8 characters for the SYSTEM, SYSTEST, and FIELD accounts.
- A list of optional VMS components you wish to install.
- A list of optional DECwindows components you wish to install.
- Information listed on Product Authorization Keys (PAKs) for your VMS license, and any system-integrated products you will run.

Once you have performed all the tasks in the checklist, go to Chapter 3 to begin the VMS installation.

2.2 The Distribution Media and the System Disk

When you install the VMS operating system, you work primarily with the distribution media and the system disk. The **distribution media** is the set of disks or tapes that the VMS operating system is supplied on. The VMS operating system is supplied on the distribution media in a format that the processor cannot readily use.

The installation procedure transfers the VMS operating system from the distribution media to your system disk and puts it in a format that the system can use. A **system disk** is the disk that contains (or will contain) the VMS operating system in a usable format.

2.3 Local Drives

A drive that is connected directly to a VAX computer is referred to as a **local drive**. For example, a magnetic tape drive connected directly to a VAX computer is referred to as a **local tape drive**.

If you have a standalone VAX computer, it is likely all the drives connected to the system are local drives.

2.4 HSC Drives

A drive that is connected to an HSC device is referred to as an **HSC drive**. For example, a magnetic tape drive connected to an HSC device is referred to as an **HSC tape drive**.

If you have a VAXcluster environment, you can have local drives, HSC drives, or both, depending on the type of VAXcluster configuration. Check with the system manager if you are not sure what types of drives you are using for the installation.

2.5 Preparing for a VMS Installation

To prepare for a VMS installation, you must perform the following tasks:

1. Make sure the hardware has been installed and checked for proper operation. For detailed information on the hardware, see the hardware manual for your VAX computer.
2. Make sure you have all the items listed on the bill of materials in the VMS distribution kit. If your kit is incomplete, notify Digital Software Supply, and request priority shipment of any missing items.
3. Before installing the VMS operating system, read the *VMS Version 5.4 New Features Manual* and *VMS Version 5.4 Release Notes* and any cover letters included with your kit.
4. If necessary, read the section ''Turning on the System'' in the VMS upgrade and installation supplement for your VAX computer and turn on power to your computer.
5. Set up your system to record the installation procedure on either a hardcopy terminal or a printer attached to the console terminal. If you do not do this, the screen messages will be lost. You will need a transcript in case there is a problem during the installation. For more information, see the VMS upgrade and installation supplement for your computer.
6. Read the ''Before Installing VMS'' section in the VMS upgrade and installation supplement for your VAX computer. This section describes tasks specific to your VAX computer that you must complete before beginning the installation.

2.5.1 Preparing for a VAXcluster Environment

If you are installing the VMS operating system in a VAXcluster environment, the installation procedure will ask you for information about your VAXcluster environment. Before proceeding, you must read the *VMS VAXcluster Manual*. If you have a clear understanding of VAXclusters before you do an installation, you are less likely to enter incorrect information during the installation. Entering incorrect information during the installation might force you to repeat the entire procedure.

Following is a list of steps to obtain the VAXcluster information you need. For a complete explanation, see the *VMS VAXcluster Manual*.

- Determine what type of configuration you want: CI-only, local area, or mixed-interconnect. These configuration types are distinguished by the interconnect device that the VAX computers in the cluster use to communicate with one another (CI, Ethernet, or both).
- Determine the **DECnet node name** and **node address** for the VAX computer on which you are installing the VMS operating system. The network or system manager determines the DECnet node name and node address for each VAX computer on the network. See your system

or network manager for this information, or see the *Guide to DECnet–VAX Networking* for more information on the DECnet node name and node address.

- Determine the appropriate allocation class value for the VAX computer on which you are installing the VMS operating system.

  During the installation procedure you will be asked for the ALLOCLASS value of the VAX computer on which you are installing the VMS operating system. For example:

  ```
  Enter a value for ALICE's ALLOCLASS parameter:
  ```

  Refer to the *VMS VAXcluster Manual* for the rules on specifying allocation class values. Note that in a mixed-interconnect VAXcluster environment, the allocation class value cannot be zero. It has to be a value between 1 and 255. This is also true for any VAX computer that is connected to a dual-pathed disk.

  When you enter the allocation class value, the installation procedure uses it to automatically set the value of the SYSGEN parameter ALLOCLASS.

- Determine whether you want a quorum disk in the cluster.
- Determine where the page and swap files will be located for the system you are installing.
- If you are going to set up either a local area or a mixed-interconnect cluster, determine which systems will be boot servers and disk servers.
- If you are going to set up either a local area or a mixed-interconnect cluster, determine the **cluster group number** and the **cluster password**. Use the following rules to determine the cluster group number and password:
  - — Cluster group number—A number in the range from 1 to 4095 or 61440 to 65535.
  - — Cluster password—Must be from 1 to 31 alphanumeric characters in length and may include dollar signs ($) and underscores (_).

2.5.2 Preparing Disk and Tape Drives

Prepare your disk and tape drives as follows:

1. Decide which drive will hold the distribution tape or disk and which drive will hold the system disk, and determine the device names for each drive. Write these names on a piece of paper. You will need this information throughout the installation. The ''Before Installing the VMS Operating System'' section in the VMS upgrade and installation supplement for your computer provides information on devices and device names.

   Do not place the distribution media in the drive at this point.

2. Follow this step only if the **source-drive** or **target-drive** is attached to an HSC. Otherwise, go to step 3.

Make sure that both the CI and HSC devices are turned on, and are on line. Obtain the HSC name from the system manager, or use the following procedure:

a. Press Ctrl/C at the HSC console terminal.

b. Enter the following command at the HSC> prompt and press RETURN:

```
HSC>  SHOW SYSTEM
```

The information displayed includes the name of the HSC. For example:

```
19-APR-1990 15:00:00.00 Boot:19-Apr-1990 11:31:11.41  Up: 51:00
Version V350                System ID: %X000000011        Name: KRUP
      .
      .
      .
DISK allocation class = 1 TAPE allocation class = 0
Start command file Disabled

SETSHO - Program Exit
```

For more information, see the *HSC User's Guide*.

3 Place a scratch disk in the drive for the system disk (unless the system disk is fixed).

4 Spin up the disk that will be the system disk but *do not* write-protect it.

2.6 Information You Need During the Installation

At different points during the installation, you will be prompted to supply information. Make sure you have this information readily available. The information you need includes the following:

- Passwords for the SYSTEM, SYSTEST, and FIELD accounts. You will be prompted to type passwords of at least 8 characters for each of these accounts.
- A list of optional VMS components you wish to install. If you want to save space on your system disk, you can choose to *not* install the *library* and *optional* VMS components. The installation procedure prompts you to specify whether you want to install each of these components. Appendix B lists the files included in these components.
- A list of DECwindows components you want to install. If you want to save space on your system disk, you can choose to *not* install certain DECwindows components. The installation procedure prompts you to specify which of the following DECwindows components you want to install: device support, applications, programming support, example files. Appendix C lists the files included in each of these components.
- Information listed on Product Authorization Keys (PAKs) for your VMS license, and any system-integrated products you will run. The installation procedure gives you the opportunity to register any licenses you have. To register your licenses, you will need to enter the information listed on the PAK for each license.

2.6 Information You Need During the Installation

Once you have completed these steps, use the checklist in Section 2.1 to make sure you have completed all the necessary tasks. Then go to Chapter 3 to begin the installation.

3 Installing the VMS Operating System and DECwindows Software

Read the cautions and restrictions in Section 3.1, then follow the instructions in this chapter to install Version 5.4 of the VMS operating system and VMS DECwindows software. At certain points you will need to refer to the VMS upgrade and installation supplement for your VAX computer.

Caution: The software installation procedure overwrites the contents of the system disk. Use the installation procedure only if your VAX computer is new, or if you want to destroy the contents of the system disk. If your system disk contains files that you want to save, you should upgrade to the new version of VMS. For an overview of the upgrade procedure, see Chapter 5.

During the installation procedure, the VMS files are transferred from the distribution kit to the system disk. You also have the option of installing VMS DECwindows software. The following tasks are described in this chapter:

1. Booting standalone BACKUP (Section 3.2)
2. Creating the system disk (Section 3.3)
3. Installing VMS DECwindows software (optional) (Section 3.4)
4. Joining a VAXcluster (optional) (Section 3.5)
5. Setting passwords (Section 3.6)
6. Applying the mandatory update (Section 3.7)
7. Registering licenses (Section 3.8)
8. Running AUTOGEN (Section 3.9)

The instructions refer to the *source-drive* and the *target-drive* where:

- The *source-drive* is the drive that holds the distribution kit.
- The *target-drive* is the drive that holds the system disk.

3.1 Cautions and Restrictions

Before you begin the VMS installation procedure, you must be aware of the following cautions and restrictions:

- You cannot install Version 5.4 of the VMS operating system on the following VAX computers and system disks:
  - MicroVAX I, VAXstation I, VAXstation 8000, and VAX 11/725 computers

— RK07, RL02, RC25, RD32, RD51, and RD52 system disks

Caution: Do not attempt to use VMSINSTAL with OPTION G to transfer VMS and VMS DECwindows save sets to another media type. If you perform an installation using such a kit, you will produce an unusable system.

- If you must move the VMS save sets from the distribution media to another location before beginning the installation, they must be in the [000000] directory on the new media, or the installation will fail.
- If your system contains Digital Small Storage Interconnect (DSSI) disks attached to a KFQSA controller, you will notice a change in the device names for DSSI disks attached to KFQSA controllers.

  When you boot standalone BACKUP, it will display the devices on your system using the *new* device naming scheme. Use the new device names throughout the installation, and after you complete the installation. For complete information on the naming scheme for DSSI disks attached to KFQSA controllers, see the *VMS Version 5.4 Release Notes*.

3.2 Booting Standalone BACKUP

Complete the following steps to boot standalone BACKUP:

1. Each piece of media in the distribution kit has a label that indicates its contents. Place the media that contains standalone BACKUP in the *source-drive*. If your kit contains standalone BACKUP on more than one piece of media, put the first piece in the *source-drive*.

   For example, there are two tape cartridges in a TK50 kit. Standalone BACKUP is on the same tape cartridge as VMS DECwindows. This tape cartridge is labeled *VMS V5.4 BIN TK50 2/2 DECWINDOWS & S/A BKUP*. To begin the installation, put this tape cartridge in the appropriate drive.

2. Follow the instructions in ''Booting Standalone BACKUP'' in the VMS upgrade and installation supplement for your VAX computer. After you boot standalone BACKUP, return to this chapter for the next step in the VMS Version 5.4 installation procedure.

3. After you boot standalone BACKUP you will see the following display:

```
%BACKUP-I-IDENT, standalone BACKUP V5.4; the date is 19-JUN-1990 15:00
$
```

Note: If you have a tape cartridge kit, remove the VMS DECwindows tape cartridge from the drive after you boot standalone BACKUP. Put the tape cartridge that contains the VMS operating system in the drive. The VMS tape cartridge is labeled *VMS V5.4 BIN TK50 1/2 VMS BINARY*.

4. Follow the instructions in Section 3.3 in this chapter to create the system disk.

3.3 Creating the System Disk

This section describes the steps for transferring the VMS files from the distribution kit to your system disk. The VMS files are stored on the distribution kit as three save sets: *required, library*, and *optional*. [1]

Note: The VMS Version 5.4 magnetic tape kit contains VMS and VMS DECwindows software on three tapes. The first tape contains the VMS *required* save set. The second tape contains the *library* and *optional* save sets. The third tape contains the VMS DECwindows software.

1 If you have not already done so, determine the device names for the *source-drive* and the *target-drive*. Write these names on a piece of paper. You will need this information throughout the installation. If either the *source-drive* or the *target-drive* are connected to an HSC, you also need the name of the HSC. (To determine the value for *hsc-name*, see Section 2.5.2.)

If you need more information about determining the device names for your system, refer to the VMS upgrade and installation supplement for your VAX computer.

2 Each piece of media in the distribution kit has a label that indicates its contents. Place the media that contains the VMS operating system in the *source-drive*. If your VMS media consists of more than one piece, insert the first piece of media in the *source-drive.* For example, if you have a TK50 tape cartridge kit, make sure the tape labeled *VMS V5.4 BIN TK50 1/2 VMS BINARY* is in the *source-drive.*

3 Enter a BACKUP command in the following format to transfer the VMS *required* save set to your system disk,

```
$ BACKUP/IMAGE/VERIFY [hsc-name]source-drive:VMS054.B/SAVE_SET [hsc-name]target-drive:
```

Substitute the appropriate device names for *hsc-name* (if appropriate), *source-drive* and *target-drive*. Make sure you put a colon (:) after each device name and that you use zeros in the device names and in VMS054.

The square brackets ([]) indicate that you need the hsc-name only if the device is connected to an HSC.

For example, if your system has the following configuration:

- A *source-drive* that is a TU80 magnetic tape drive with a device name of MUA0.
- A *target-drive* that is an RA60 disk drive with a device name of DJA1.

Enter the following command and press the Return key:

```
$ BACKUP/IMAGE/VERIFY MUA0:VMS054.B/SAVE_SET DJA1:
```

[1] If the distribution media is a disk, the VMS save sets are in the [000000] directory. They must be in this directory for the installation procedure to work.

If your system has the following configuration:

- A *source-drive* that is a tape cartridge drive with a device name of MUC6.
- A *target-drive* that is an RA81 disk drive with a device name of DUA1 and an HSC name of YOURS.

Enter the following command and press Return:

```
$  BACKUP/IMAGE/VERIFY MUC6:VMS054.B/SAVE_SET YOURS$DUA1:
```

4 The procedure transfers the VMS *required* save set from the distribution kit to the system disk.[2] The amount of time this takes varies depending upon the type of system and distribution media you have. During the process the procedure displays the following message:

```
%BACKUP-I-STARTVERIFY, starting verification pass
```

This message indicates that the *required* save set has been transferred to the system disk and the files are being checked for errors. If, instead, you see a message similiar to the following, you might have the wrong volume of the distribution kit in the source drive:

```
% BACKUP-F-OPENIN, error opening  MUA0:[000000]VMS054.B; as
  input
- SYSTEM-W-NOSUCHFILE, no such file
    .
    .
    .
If you do want to perform another standalone BACKUP operation,
ensure the standalone BACKUP volume is online and ready.
Enter "YES" to continue:
```

If the procedure displays the previous messages, remove the distribution volume from the drive, and replace it with the correct volume as described in step 2. Then type YES, press Return, and go back to step 3.

When standalone BACKUP is done checking the files, the procedure displays the following message:

```
%BACKUP-I-PROCDONE, Operation completed. Processing finished at 19-JUN-1990 15:00
If you do not want to perform another standalone BACKUP operation,
use the console to halt the system.

If you do want to perform another standalone BACKUP operation,
ensure the standalone application volume is online and ready.
Enter "YES" to continue:
```

5 Halt the system and boot the new system disk, as described in the VMS upgrade and installation supplement for your VAX computer. After you boot the new system disk, return to this manual for the next step in the VMS Version 5.4 installation procedure.

[2] The BACKUP command creates a system disk that includes a set of volume parameters provided by Digital, including CLUSTER_SIZE (disk access scheme). For more information, see the VMS upgrade and installation supplement for your VAX computer.

6 When the boot is complete, the procedure displays a message and prompts you to enter the date and time. Enter the date and time using the 24-hour clock format and press Return. For example:

```
  VAX/VMS Version BI54-xxx Major version id = 1 Minor version id = 0

     VAX/VMS V5.4 Installation Procedure

                            Model:  VAX 6000-410
                    System device:  RA60 - DJA3
                      Free blocks: 359916
                         CPU type: 11-02

* Please enter the date and time (DD-MMM-YYYY HH:MM):  19-JUN-1990 15:00
%SET-I-NEWAUDSERV, identification of new audit server process is 00000027
%LICENSE-F-EMTLDB, license database contains no license records
%LICENSE-E-NOAUTH, DEC VAX-VMS use is not authorized on this node
-%LICENSE-F-NOLICENSE, no license is active for this software product
-%LICENSE-I-SYSMGR, please see your system manager
Startup processing continuing...

%%%%%%%%%%  OPCOM 19-JUN-1990 15:00:00.00  %%%%%%%%%%
    .
    .
    .
```

Note: The procedure displays error messages stating that the VMS license is not registered. At this point, you can ignore these messages. You will be given the opportunity to register licenses later in the procedure. For more information about registering your licenses, see the *VMS License Management Utility Manual*.

You can also ignore the following messages:

```
%SYSGEN-W-OPENIN, error opening SYS$SYSROOT:[SYSEXE]VEDRIVER.EXE; as input
%SYSGEN-E-FNF, file not found
```

7 The procedure prompts you to specify the volume label of the system disk. A volume label is the name the VMS operating system uses to refer to the system disk. You can press Return to accept the default volume label, VAXVMSRL054, or you can enter a volume label of your choice. The volume label can be 1 to 12 characters long; do not use spaces. For example:

```
If this system disk is to be used in a cluster with multiple system
disks, then each system disk must have a unique volume label. Any
nodes having system disks with duplicate volume labels will fail to
boot into the cluster.

You may indicate a volume label of 1 to 12-characters in length. If you
wish to use the default name of VAXVMSRL054 just press Return in response
to the next question.

* Enter the volume label for this system disk [VAXVMSRL054]:  SYSDSK
```

8 The procedure prompts you to specify which drive holds the distribution kit. Enter the device name of the *source-drive*.

For example, if the *source-drive* is a tape cartridge drive with a device name of MUA0 type "MUA0" and press Return.

```
* Enter the name of the drive holding the VMS distribution media:  MUA0
* Is the VMS media ready to be mounted? [N]  YES
%MOUNT-I-MOUNTED, VMS054       mounted on _MUA0:
```

9 The procedure displays information about VMS and DECwindows files that are optional, and the remaining amount of disk space. The information is similiar to the following:

```
Select Optional software you wish to have installed. You can choose to
Install one or more of the following VMS and/or DECwindows components:

     o VMS library                    - 30000 blocks
     o VMS optional                   - 13000 blocks
     o DECwindows device support      - 13500 blocks
     o DECwindows applications        - 12000 blocks
     o DECwindows programming support - 26500 blocks
     o DECwindows example files       -  3000 blocks

 Space remaining: 359715 blocks
```

You must indicate which of the options you want to install.

NOTE: If you do not install the optional VMS and DECwindows components at this time, you can install them after the installation by using the VMSTAILOR and DECW$TAILOR utilities. Please note that, for TK50 tape cartridges, it is quicker to install VMS and DECwindows software during the installation than to add software after the upgrade. For information on using VMSTAILOR and DECW$TAILOR, see Chapter 12.

To select the options you want to install, follow the instructions displayed by the procedure. For example:

```
* Do you want to install the VMS library files? (Y/N)
```

10 The contents of the VMS *library* save set are listed in Appendix B. If you want to install the VMS *library* files, type Y (for YES) and press Return. If you do *not* want to install the VMS *library* files, type N (for NO) and press Return. The procedure displays the following message:

```
* Do you want to install the VMS optional files? (Y/N)
```

11 The contents of the VMS *optional* save set are listed in Appendix B. If you want to install the VMS *optional* files, type Y and press Return. If you do *not* want to install the VMS *optional* files, type N and press Return.

The procedure displays some messages about DECwindows software.

```
    You can provide DECwindows support now, or you can use the
    DECW$TAILOR utility to provide or remove DECwindows support
    after the installation.

    NOTE: Some media, TK50s in particular, can be very slow when
    tailoring on files. You might want to provide DECwindows
    options now and tailor off unwanted files later.

* Do you want to provide optional DECwindows support? (Y/N)
```

12 You must indicate if you wish to install DECwindows software.

If you *do not* want to install VMS DECwindows software, type N, press Return, and go to step 13. Note that VMS DECwindows software is useful, even on a nonworkstation VAX computer, because you can run VMS DECwindows applications on a nonworkstation VAX computer and display them on a workstation.

If you want to install VMS DECwindows software, follow steps **a** and **b**:

a. Type Y and press Return. The procedure displays the following message:

```
You can install either the complete DECwindows kit or selected portions
of it. A complete installation requires 56000 blocks. If you
want to perform a partial installation, you can choose to install one
or more of the following DECwindows components:

    o DECwindows device support      -  13500 blocks
    o DECwindows applications        -  12000 blocks
    o DECwindows programming support -  26500 blocks
    o DECwindows example files       -   3000 blocks

* Do you want to install complete DECwindows support? (Y/N)
```

b. To install the complete DECwindows kit, type Y and press Return. The procedure displays the amount of remaining disk space, and the VMS DECwindows options that will be provided.

To install selected portions of the VMS DECwindows kit, type N and press Return. The procedure displays information about the amount of space you need on the system disk for each option. As you choose your options, the procedure calculates the amount of space that would be left on the system disk.

Before you select the VMS DECwindows options that you want to install, consider the following:

- Device support files—If you are installing VMS DECwindows on a VAXstation or in a cluster that includes VAXstations, you need these files. Otherwise you do not need them. Please note that if you select the device support files, you automatically get applications files and 75 dots-per-inch (75 DPI) video font files, and you will be prompted to indicate if you want to install 100 dots-per-inch (100 DPI) video font files.
- Applications files—VMS DECwindows provides a variety of applications including the following: Bookreader, Calculator, Calendar, Cardfiler, CDA Document Viewer, Clock, EVE Text Editor, FileView, Mail, Notepad Text Editor, Paint Graphics Editor, DECterm, the Session Manager, and the Window Manager. If you are installing VMS DECwindows on a VAXstation, you need these files. They are also useful on a nonworkstation VAX computer.
- 100 dots-per-inch (100 DPI) video font files—If you have a workstation monitor capable of displaying 100 dpi (for example, a VR150, VR160, or VR295), selecting 100 dpi fonts will give you a more readable display when you run VMS DECwindows applications. If your workstation monitor cannot display 100 dpi fonts (for example, a VR260 or VR290), you can save disk space by not selecting this option. For more information on the capabilities of your particular monitor, see the owner's guide for your monitor.

If you select the device support option you will be prompted to specify if you also want to select the 100 DPI files.

- Programming support files—The VMS DECwindows programming support files are needed only to develop applications and are not required to run VMS DECwindows applications. Support files are supplied for the following VAX programming languages: Ada, BASIC, BLISS, C, FORTRAN, MACRO, Pascal, and PL/1. In addition, support files are supplied for programming in the C language using conventions and standards developed at the Massachusetts Institute of Technology.
- DECwindows example files—The VMS DECwindows example files provide examples of files used to develop various DECwindows applications.

13 The procedure gives you the opportunity to change your mind before proceeding by displaying the following message:

```
* Is this correct? (Y/N)
```

If you answer YES to this prompt, the installation procedure begins restoring the VMS *library* save set (if you elected to install it). If you answer NO to this prompt, you can respecify the VMS and DECwindows options that you want to install.

14 If you elected to install the VMS *library* and *optional* save sets, the procedure displays the following messages:

```
Restoring VMS library save set ...
%BACKUP-I-STARTVERIFY, starting verification pass

Restoring VMS optional save set ...
%BACKUP-I-STARTVERIFY, starting verification pass
```

You can use the VMSTAILOR utility to remove the optional VMS files that you do not need after you finish the installation. For more information on removing files with VMSTAILOR see Chapter 12.

15 If you are installing VMS DECwindows software, go to Section 3.4.

If you are not installing VMS DECwindows software, go to Section 3.5.

3.4 Installing VMS DECwindows Software

If you are installing VMS DECwindows software, complete the following steps:

1 Remove the tape or disk that contains the VMS operating system from the *source-drive* when you see a message similar to the following:

```
You may now remove the VMS distribution kit from MUA0:.

Load the DECwindows distribution kit.
```

2 Put the tape or disk that contains the VMS DECwindows software in the *source-drive*.

Note: If you have an RL02 kit, VMS DECwindows is on a magnetic tape. If you have an RX33 kit, VMS DECwindows is on a tape

cartridge. Check the labels and put the appropriate piece of media in the appropriate drive.

3 Type the device name of the *source-drive* and press Return. For example:

```
* Enter name of the drive holding the DECwindows distribution media:   MUA0
* Is the DECwindows media ready to be mounted? [N]  YES
```

4 The procedure installs the VMS DECwindows options that you specified and displays messages similar to the following:

```
%MOUNT-I-MOUNTED, DECW54 mounted on _MUA0:

    Restoring DECwindows applications save set ...
%BACKUP-I-STARTVERIFY, starting verification pass

    Restoring DECwindows device support save set ...
%BACKUP-I-STARTVERIFY, starting verification pass

    Restoring DECwindows 75DPI fonts save set ...
%BACKUP-I-STARTVERIFY, starting verification pass

    Restoring DECwindows 100DPI fonts save set ...
%BACKUP-I-STARTVERIFY, starting verification pass

    Restoring DECwindows programming support save set ...
%BACKUP-I-STARTVERIFY, starting verification pass

    Restoring DECwindows examples save set ...
%BACKUP-I-STARTVERIFY, starting verification pass
```

The VMS DECwindows files are listed in Appendix C. You can use the DECwindows tailoring utility, DECW$TAILOR, to remove the VMS DECwindows files you do not need after you finish the installation. For more information on removing files with DECW$TAILOR, see Chapter 12.

5 If you installed VMS DECwindows options, the procedure displays a message similar to the following:

```
You may now remove the DECwindows distribution kit from MUA0:
```

Remove the distribution media from the drive.

6 Go to Section 3.5 to continue the installation procedure.

3.5 Joining a VAXcluster

The installation procedure now asks you whether you are installing the VMS operating system in a VAXcluster environment. If you are installing a VAXcluster, the procedure asks a series of questions about your configuration. You might need to refer to the *VMS VAXcluster Manual* to answer these questions.

1 The procedure displays the following messages:

```
Creating [VMS$COMMON] directory tree

In a cluster, you can run multiple systems sharing all files except
PAGEFILE.SYS, SWAPFILE.SYS, SYSDUMP.DMP and VAXVMSSYS.PAR.

* Will this node be a cluster member? (Y/N)
```

If you are installing a standalone system, type N and press Return.

If you are installing a VAXcluster, type Y and press Return. If you answer YES to the VAXcluster question, you must have a VAXcluster license.

2 If you are installing the VMS operating system in a VAXcluster environment, go to step 3.

If you are installing the VMS operating system on a standalone system, go to step 5.

3 The procedure displays the following message:

```
Now configuring system to be a cluster member.
```

4 The installation procedure displays several questions about the configuration of your VAXcluster. Your answers to these questions depend on whether you want to install a CI-only, local area, or mixed interconnect VAXcluster. Before answering these questions, you must determine which type of cluster you want to install. For a complete description of VAXcluster types, see the *VMS VAXcluster Manual*.

Table 3–1 lists the questions you are asked if you want a CI-only VAXcluster. Table 3–2 lists the questions you are asked if you want a local area VAXcluster or a mixed-interconnect VAXcluster. Typical responses are explained in the tables. Refer to the appropriate table to enter the information for your VAXcluster configuration. When you have answered all the questions, go to Section 3.6.

Table 3–1 Installation Questions for CI-Only VAXclusters

Question	How to Respond
Will this node be a cluster member (Y/N)?	Enter Y.
What is the node's DECnet node name?	Enter the DECnet node name—for example, KRUP. The DECnet node name may be from 1 to 6 alphanumeric characters in length and cannot include dollar signs or underscores.
What is the node's DECnet node address?	Enter the DECnet node address—for example, 2.2.
Will the Ethernet be used for cluster communications (Y/N)?	Enter N. A CI-only VAXcluster does not use the Ethernet for cluster communications.

(continued on next page)

Table 3–1 (Cont.) Installation Questions for CI-Only VAXclusters

Question	How to Respond
Will KRUP be a disk server (Y/N)?	Enter Y if you want local disks to be served to the cluster. If you enter Y, the procedure will ask if the node will serve HSC disks. Enter N to this second question. HSC disks are available to all nodes in a CI-cluster. Refer to the *VMS VAXcluster Manual* for information on served cluster disks.
Enter a value for KRUP's ALLOCLASS parameter:	If the system is connected to a dual-ported disk, enter the appropriate allocation class value (it must be a value between 1 and 255). Otherwise, enter 0. For information on selecting the ALLOCLASS parameter, see the *VMS VAXcluster Manual*.
Does this cluster contain a quorum disk (Y/N)?	Enter Y or N, depending on your configuration. If you enter Y, the procedure asks for the name of the quorum disk. Enter the device name of the quorum disk. Refer to the *VMS VAXcluster Manual* for information on quorum disks.

Table 3–2 Installation Questions for Local Area and Mixed-Interconnect VAXclusters

Question	How to Respond
Will this node be a cluster member (Y/N)?	Enter Y.
What is the node's DECnet node name?	Enter the DECnet node name—for example, KRUP. The DECnet node name may be from 1 to 6 alphanumeric characters in length and may not include dollar signs or underscores.
What is the node's DECnet node address?	Enter the DECnet node address—for example, 2.2.
Will the Ethernet be used for cluster communications (Y/N)?	Enter Y. The Ethernet is required for cluster (SCS internode) communications in local area and mixed-interconnect VAXclusters.
Enter this cluster's group number:	Enter a number in the range from 1 to 4095 or 61440 to 65535.
Enter this cluster's password:	Enter the cluster password. The password must be from 1 to 31 alphanumeric characters in length and may include dollar signs and underscores.
Re-enter this cluster's password for verification:	Re-enter the password.

(continued on next page)

Table 3–2 (Cont.) Installation Questions for Local Area and Mixed-Interconnect VAXclusters

Question	How to Respond
Will KRUP be a disk server (Y/N)?	Enter Y. In local area and mixed-interconnect VAXclusters, the system disk is always served to the cluster. Refer to the *VMS VAXcluster Manual* for information on served cluster disks.
Will KRUP serve HSC disks (Y)?	Enter a response appropriate for your configuration.
Enter a value for KRUP's ALLOCLASS parameter:	Enter the appropriate allocation class value. If you have a mixed-interconnect VAXcluster, the value must be between 1 and 255; you cannot enter 0.
Does this cluster contain a quorum disk (Y/N)?	Enter Y or N, depending on your configuration. If you enter Y, the procedure asks for the name of the quorum disk. Enter the device name of the quorum disk. Refer to the *VMS VAXcluster Manual* for information on quorum disks.

5 If you installed VMS DECwindows and indicated that the node is not part of a VAXcluster, the procedure displays the following message:

```
* Do you want DECwindows as the default windowing system? (Y/N)
```

Depending on your preference, type Y or N and press Return. If you type Y, the procedure sets the SYSGEN parameter WINDOW_SYSTEM to one for DECwindows. If you type N, the procedure sets the SYSGEN parameter WINDOW_SYSTEM to zero. For information on the WINDOW_SYSTEM parameter, see the *VMS System Generation Utility Manual*.

6 Go to Section 3.6.

3.6 Setting Passwords

Next, the installation procedure prompts you to set new passwords for certain accounts and uses this information to create a rights database.

1 The procedure asks you for new passwords for the SYSTEM, SYSTEST, and FIELD accounts. Passwords must be at least 8 characters in length; they do not appear on the display. Press Return after you enter each one. After you enter the passwords, the procedure checks each one to make sure it meets the requirements for a good password. For example:

```
Now we will ask you for new passwords for the following accounts:
                SYSTEM, SYSTEST, FIELD

Passwords must be a minimum of 8 characters in length. All passwords
will be checked and verified. Any passwords that can be guessed easily
will not be accepted.
```

```
* Enter password for SYSTEM:
* Re-enter for verification:
%UAF-I-MDFYMSG, user record(s) updated
%VMS-I-PWD_OKAY, account password for SYSTEM verified

* Enter password for SYSTEST:
* Re-enter for verification:
%UAF-I-MDFYMSG, user record(s) updated
%VMS-I-PWD_OKAY, account password for SYSTEST verified

     The SYSTEST_CLIG account will be disabled. You must re-enable
     it before running UETP but do not assign a password.
* Enter password for FIELD:
* Re-enter for verification:
%UAF-I-MDFYMSG, user record(s) updated
%VMS-I-PWD_OKAY, account password for FIELD verified
```

If you do not get password errors, go to step 2.

If you enter a password incorrectly or if the password is too easy to guess, the procedure displays error messages similar to the following:

```
%VMS-I-PWD_INVALID, account password for SYSTEST is invalid
%VMS-I-PWD_WEAK, password is too easy to guess

Because of the preceding error, you must take action to secure this account.
You must either disable this account, change its password, or do both.
```

When the procedure asks if you want to disable the account, type N and press Return. When the procedure asks if you want to enter a new password, type Y and press Return. Then enter a new password. For example:

```
Do you want to disable this account (Y/N)?  N
Do you want to change the account password (Y/N)?  Y
You must now select a new primary password for the SYSTEST account. The
password you select must be at least 8 characters in length and may not
be the same as the name of the account:

New password:
Verification:
%UAF-I-MDFYMSG, user record(s) updated
%VMS-I-PWD_SET, primary password for account SYSTEST set
```

2 After you have entered the passwords, the procedure creates your RIGHTS database and displays the following message:

```
Creating RIGHTS database file, SYS$SYSTEM:RIGHTSLIST.DAT
Ignore any "-SYSTEM-F-DUPIDENT, duplicate identifier" errors
      .
      .
      .
```

3 After the procedure creates the RIGHTS database, it displays the following messages:

```
After the installation finishes, you may want to do one or more of the
following tasks:

o DECOMPRESS THE SYSTEM LIBRARIES - For space considerations, many of
the system libraries are shipped in a data-compressed format.  If you
have enough disk space, you may decompress them for faster access. To
data expand the libraries, type:

    $ @SYS$UPDATE:LIBDECOMP.COM
```

```
If you choose not to decompress these libraries you will experience
slower response with the HELP and LINK commands.

o BUILD A STANDALONE BACKUP KIT - You can build a standalone backup kit
using the procedure described in your VMS upgrade and installation supplement
which is supplied with your VAX computer.

o TAILOR THE SYSTEM DISK - You may want to review the files provided or
not provided during this installation. IF you find there are files
you want to remove from the system disk (TAILOR OFF) or files you want
to add (TAILOR ON), then use the following utilities to perform the
desired tailoring.

    VMS tailoring:          $ RUN SYS$UPDATE:VMSTAILOR

    DECwindows tailoring:   $ RUN SYS$UPDATE:DECW$TAILOR
```

3.7 Applying the Mandatory Update

After displaying messages about post-installation tasks, the procedure displays the following message:

```
Continuing with VAX/VMS V5.4 installation procedure.

Configuring all devices on the system...

You must now install the Mandatory Update, which can be found
on a separate distribution volume.
.
.
.
```

Follow these steps to apply the mandatory update:

1 The procedure asks you for the device name of the drive that contains the mandatory update. Type the device name of the drive where you will put the disk or tape that contains the mandatory update.

For example, if the mandatory update is on a console drive named CSA1, enter the following and press the Return key:

```
*Where will the distribution volumes be mounted:  CSA1
```

The procedure displays the following message:

```
Please mount the first volume of the set on CSA1:.
*Are you ready?
```

2 Place the disk or tape labeled VMS V5.4 MANDATORY UPDATE in the appropriate drive. When you are ready to continue, type Y and press Return. The procedure displays the following series of messages:

```
%MOUNT-I-MOUNTED, VMSMUP mounted on _CSA1:

The following products will be processed:

   VMSMUP V5.4

        Beginning installation of VMSMUP V5.4 at 15:00

%VMSINSTAL-I-RESTORE, Restoring product save set A ...
        .
        .
        .
```

3 If there is more than one piece of disk or tape containing the mandatory update, you will be asked to switch them at some point during the procedure. Read the screen displays. When the procedure finishes, it displays a message similar to the following:

```
Installation of VMSMUP V5.4 completed at 15:00

VMSINSTAL procedure done at 15:02
```

3.8 Registering Licenses

After you apply the mandatory update, the procedure gives you the opportunity to register your software licenses.

The VMS license permits you to use the VMS operating system. You must register this license.

After you register the VMS license, you must register the license for any of the following system integrated products you have purchased:

VAXclusters
DECnet–VAX
RMS Journaling
Volume Shadowing

If you do not register your licenses at this point, you must register them immediately after the installation, before performing the other post-installation tasks.

1 The installation procedures displays the following message:

```
   If you have any Product Authorization Keys (PAKS) to register, you may
   register them now.

* Do you want to register any Product Authorization Keys? (Y/N)
```

2 If you want to register a license, type Y and press Return. You must have a copy of your Product Authorization Key (PAK) for each license you will register.

If you do not want to register the licenses at this time, type N, press Return, and go to Section 3.9.

3 The procedure displays the following message:

```
    VMS License Management Utility Options:

        1. Register a Product Authorization Key
        2. Amend an existing Product Authorization Key
        3. Cancel an existing Product Authorization Key
        4. List Product Authorization Keys
        5. Modify an existing Product Authorization Key

        9. Exit this procedure

    Type '?' at any prompt for a description of the information
    requested.

Enter one of the above choices [1]
```

4 Refer to the description of VMSLICENSE.COM in the *VMS License Management Utility Manual* for step-by-step instructions to register licenses.

3.9 Running AUTOGEN

The remaining portion of the installation procedure runs AUTOGEN to evaluate your hardware configuration and estimate typical workloads. AUTOGEN then sets system parameters, the sizes of page, swap, and dump files, and the contents of VMSIMAGES.DAT. When AUTOGEN finishes and you reboot, the installation procedure is complete.

1 The installation procedure displays the following series of messages:

```
Running AUTOGEN to compute new SYSGEN parameters.

An attempt may be made to resize the pagefile or swapfile. If
there is insufficient room on the disk, the recommended size is
displayed with a message that the file should be created or
extended manually by the system manager later on.

Running  AUTOGEN - Please wait.
     .
     .
     .
```

If you need to create or modify the size of the pagefile or swapfile, use the SYSGEN utility. For more information, see the *Guide to Setting Up a VMS System*.

2 After AUTOGEN finishes, the procedure displays a series of shutdown messages that begins like this:

```
The system is shutting down to allow the system to boot with the
generated site-specific parameters and installed images.

The system will automatically reboot after the shutdown and the upgrade
will be complete.

     SHUTDOWN -- Perform an Orderly System Shutdown
```

3 After the system shuts down, the procedure tries to reboot the new system disk. If the system does not reboot automatically, reboot it manually.

For example, suppose you have a VAX–11/780 and the system disk is on an RA60 disk drive with a unit number of one. Enter the following command and press Return:

```
>>>  B DU1
```

For information on booting, refer to the VMS upgrade and installation supplement for your VAX computer. Then return to this manual.

4 After the system reboots, the procedure displays the following message:

```
   VAX/VMS Version V5.4 Major version id = 1 Minor version id = 0
************************************************************

VAX/VMS V5.4

You have SUCCESSFULLY installed the VMS operating system.
```

```
The system is now executing the STARTUP procedure. Please
wait the completion of STARTUP before logging in to the
system.

******************************************************************

The VAX/VMS system is now executing the system startup procedure.

%%%%%%%%%%  OPCOM 19-JUN-1990 15:00.00.00  %%%%%%%%%%
   .
   .
   .
```

Note: If you did not register your VMS license as described in Section 3.8, the procedure displays warning messages that the VMS license must be registered. Be sure to register this license when the installation procedure finishes. For more information, see the *VMS License Management Utility Manual*.

Finally, the procedure displays informational messages and accounting information. For example:

```
The VAX/VMS system is now executing the system startup commands.

%SET-I-INTSET, login interactive limit = 64, current interactive value = 0
  19-JUN-1990 15:00:00.00
  SYSTEM       job terminated at 19-JUN-1990 15:00:00.00

Accounting information:
 Buffered I/O count:       859      Peak working set size:        565
 Direct I/O count:         478      Peak virtual size:           2570
 Page faults:             5003      Mounted volumes:                0
 Charged CPU time:  0 00:00:55.33   Elapsed time:       0 00:01:31.24
```

At this point the VMS operating system is running.

5 Press the Return key. The system asks you for the user name and password. Log into the SYSTEM account so that you can perform certain post-installation tasks. For example:

```
     Welcome to VAX/VMS V5.4

Username:  SYSTEM
Password:
     .
     .
     .
     Welcome to VAX/VMS Version V5.4
```

If you forget the password, follow the instructions for performing an emergency startup in the *Guide to Setting Up a VMS System*.

6 There are several things you must do before you can use the system. For complete information, see Chapter 4.

4 After Installing VMS

After you have installed the VMS operating system, you need to perform several important tasks to prepare the system for operation. This chapter describes these tasks in the order in which you perform them.

4.1 Post-Installation Checklist

If you are an experienced system manager, use the following checklist to make sure you perform the necessary tasks. Less experienced system managers are advised to follow the step-by-step instructions beginning in Section 4.2, and to use the following checklist when they have completed the post-installation tasks.

- ☐ Register your licenses, if you did not do so during the installation procedure (described in the *VMS License Management Utility Manual*).
- ☐ Build a standalone BACKUP kit (described in the VMS upgrade and installation supplement for your computer).
- ☐ Back up the system disk as a safeguard before customizing the system (described in the VMS upgrade and installation supplement for your computer).
- ☐ Customize the system (described in Section 4.4).
- ☐ Configure and start DECnet–VAX (described in Section 4.5).
- ☐ Create a transaction log file or disable DECdtm software, depending on whether or not you want to use DECdtm services (described in Section 4.6).
- ☐ Run the VMS User Environment Test Package (UETP) to test the system (described in Chapter 10) .
- ☐ Decompress the system libraries using LIBDECOMP.COM (described in Section 4.8).
- ☐ Set up the system messages help library (described in Section 4.9).
- ☐ Remove unwanted files from the system disk using the VMS and DECwindows tailoring utilities (described in Chapter 12).
- ☐ Customize VMS DECwindows (described in Chapter 11).
- ☐ Back up the system disk after you have customized it (described in the VMS upgrade and installation supplement for your computer).
- ☐ After the system has been running for at least 24 hours, run AUTOGEN to collect feedback information (described in Section 4.13).

4.2 Registering Your Licenses

The installation procedure gave you the opportunity the register any software product licenses. If you did not register your VMS license at that time, you must do so before you can use the VMS operating system. You must also register the licenses for any of the following system integrated products you have purchased:

VAXclusters
DECnet–VAX
RMS Journaling
Volume Shadowing

For step-by-step instructions on registering licenses, see the *VMS License Management Utility Manual*.

4.3 Backing Up Your System Disk

Now that you have spent time installing the operating system, you should protect your work by making a backup copy of the system disk in case you have any problems while customizing it. Digital recommends that you perform the following operations:

- Make a standalone BACKUP kit
- Back up the system disk

You should also back up your system disk after you have customized it.

For complete information on these operations, see the VMS upgrade and installation supplement for your VAX computer.

4.4 Customizing the System

You can customize the system to meet your site-specific needs. In addition, if your VAX computer is part of a VAXcluster environment, you must prepare the cluster environment and build the cluster.

For instructions on customizing the system, read the following documentation (in the order given):

1. Read the chapter on booting in the VMS upgrade and installation supplement for your VAX computer. This chapter explains the different ways to boot the system. It also tells you how to shut down the system.
2. If the computer is part of a VAXcluster environment, read the *VMS VAXcluster Manual* for further information on setting up a cluster.
3. Read the *Guide to Setting Up a VMS System* for instructions on customizing and using your system. You will find information on the following tasks:
   a. Editing the template files SYCONFIG.COM, SYLOGICALS.COM, SYLOGIN.COM, and SYSTARTUP_V5.COM
   b. Setting up user accounts
   c. Adjusting system parameters

4 Read Section 4.5 in this chapter, and perform the steps to configure and start DECnet–VAX.

5 Read Section 4.6 in this chapter, and perform the steps appropriate for your configuration.

4.5 Configuring and Starting DECnet–VAX

If you plan to run DECnet–VAX, you must do the following:

1 Once you have registered the license for DECnet–VAX, execute the interactive command procedure SYS$MANAGER:NETCONFIG.COM to automatically configure your system for networking. See the *Guide to DECnet–VAX Networking* for instructions on using NETCONFIG.COM.

2 Edit the commands in SYS$COMMON:[SYSMGR]SYSTARTUP_V5.COM that pertain to networking, so that DECnet starts automatically when your system boots. Choose one of the following commands to start the network and remove the comment delimiter (!) from that command:

```
$! IF F$SEARCH("SYS$SYSTEM:NETACP.EXE") .NES. "" THEN @SYS$MANAGER:STARTNET
$! IF F$SEARCH("SYS$SYSTEM:NETACP.EXE") .NES. "" THEN SUBMIT SYS$MANAGER:STARTNET.COM
```

Both of the previous commands perform the same task. However, the first command executes STARTNET.COM and delays further processing until the procedure completes; the second submits STARTNET.COM to a batch queue and continues executing SYSTARTUP_V5.COM.

3 If you plan to run both DECnet–VAX and VMS DECwindows, you must also edit SYS$COMMON:[SYSMGR]SYSTARTUP_V5.COM to add a comment delimiter (!) immediately following the dollar-sign ($) in the following command:

```
$ DEFINE DECW$IGNORE_DECNET TRUE
```

If you are not going to start DECnet–VAX or have not yet started it, this command tells DECwindows not to wait for DECnet–VAX.

4.6 Customizing or Disabling DECdtm Services

By default, processes for DECdtm services are started when a full VMS boot is executed. The DECdtm process TP_SERVER then checks for the existence of a transaction log file on the system and continues checking every 15 seconds.

If you plan to run DECdtm software you must create a transaction log file before transactions can be started. To create a transaction log file, you use the LMCP utility. Refer to the LMCP section in the *VMS Version 5.4 New Features Manual* for information about creating a log file.

If you do not want to run DECdtm software, you can prevent the startup of DECdtm processes by defining the system-wide logical name SYS$DECDTM_INHIBIT in the SYS$MANAGER:SYLOGICALS.COM command procedure. You can define SYS$DECDTM_INHIBIT to be any string. For example:

```
$ DEFINE/SYSTEM/EXEC SYS$DECDTM_INHIBIT "yes"
```

For information on the SYLOGICALS.COM command procedure, see *Guide to Setting Up a VMS System.*

4.7 Testing the System

Run the User Environment Test Package (UETP) to verify the installation. For complete information, see Chapter 10. Note that UETP needs at least 1200 free blocks on the system disk.

4.8 Decompressing the System Libraries

Decompressing the system libraries gives the system faster access to them. The decompressed libraries require approximately 8500 additional blocks of disk space. To find out how much disk space you have, enter the following command and press the Return key:

```
$  SHOW DEVICE SYS$SYSDEVICE
```

If you have at least 8500 free blocks on the disk, you can decompress the libraries. The decompression process takes approximately a half hour. Log into the SYSTEM account, enter the following command, and press Return:

```
$  @SYS$UPDATE:LIBDECOMP.COM
```

4.9 Setting Up the System Messages Help Library

For VMS Version 5.4, you can install and access an optional online Help version of the *VMS System Messages and Recovery Procedures Reference Manual*. Because this is a large file, it is *not* included as part of the default root library, SYS$HELP:HELPLIB.HLB. Instead, it is provided as a separate file, named SYS$HELP:SYSMSGHELP.HLB. You can view the system messages by accessing the file in one of three ways:

- Use the /LIBRARY qualifier with the HELP command. For example:

  ```
  $  HELP/LIBRARY=SYS$HELP:SYSMSGHELP.HLB ERRORS ACCVIO
  ```

- Or, define a logical name that instructs the Help Facility to search the new help library when it does not find the specified topic in the VMS root help library. For example:

  ```
  $  DEFINE HLP$LIBRARY DISK$2:[QUAIL]SYSMSGHELP
  $  HELP ERRORS DISMAL
  ```

In this example, the DEFINE statement creates a logical name for the help library that the Help Facility will search after it has searched the root library, SYS$HELP:HELPLIB.HLB. In this example, the help library is located in DISK$2:[QUAIL]; you should specify the appropriate device and directory for your system.

First, the Help facility searches the root library for ERRORS. If it does not find ERRORS,[1] it searches the library defined by HLP$LIBRARY, until it finds ERRORS and displays the appropriate information. For information on defining logical names and search patterns for the Help facility, see the HELP command in the *VMS DCL Dictionary*.

- Or, using the VMS Librarian Utility, you can extract the ERRORS module from SYSMSGHELP.HLB and insert it into the default root help library HELPLIB.HLB. This allows direct access without using extra HELP qualifiers or logical names. For more information, see the *VMS Librarian Utility Manual*.

The system messages help library is in compressed format. Decompressing the library gives you faster access to it but requires an additional 1600 blocks of disk space. To decompress the library, enter a command similiar to the following:

```
$  LIBRARY/DATA=EXPAND/OUTPUT=device:[directory]SYSMSGHELP.HLB -
_$  device:[directory]SYSMSGHELP.HLB
```

In this example, *device* is the name of the device where the file is located, and *directory* is the name of the directory.

Note: The system messages help library is not decompressed when you execute the LIBDECOMP.COM procedure described in Section 4.8.

The system messages help library can be deleted without affecting other help libraries. You can also add or delete this library using the VMS tailoring utility (VMSTAILOR).

4.10 Removing Unwanted Files

To remove unwanted VMS files from the system disk, use VMSTAILOR. For example, if you are not running DECnet–VAX, use VMSTAILOR to remove the files associated with DECnet–VAX. For complete information on using VMSTAILOR to remove optional VMS software see Chapter 12.

To remove unwanted DECwindows files from the system disk, use DECW$TAILOR. For example, if you do not plan to use the DECwindows programming support, you can remove those files. For complete information on removing DECwindows files with DECW$TAILOR see Chapter 12.

[1] Previous versions of HELPLIB.HLB provided information about system messages format under the name ERROR. This information is now named FORMAT_OF_ERROR.

4.11 Customizing DECwindows

After you register the VMS license and reboot the system, DECwindows starts automatically (if you installed it).

If you have installed VMS DECwindows software, there are several steps that you must take to customize the VMS DECwindows environment. Chapter 11 describes these steps. After you finish customizing VMS DECwindows software, return to this chapter.

4.12 Backing Up the Customized System Disk

Now that you have spent time installing and customizing the operating system, protect your work by making a backup copy of the system disk.

For complete information on these operations, see the VMS upgrade and installation supplement for your VAX computer. Once you have made a backup copy of the system disk, install any software products that you have purchased. Follow the directions given in the software product manuals.

4.13 Running AUTOGEN

After 24 hours of operation, run AUTOGEN in feedback mode and reboot the system. Run AUTOGEN in this way again two workdays later.

Note: For the VAX 9000 computer, AUTOGEN's initial parameter calculations are conservative. To obtain parameter values that match your system workload, you may need to run AUTOGEN in feedback mode a number of times. For more information, see the VAX 9000 notes in the system manager chapter of the *VMS Version 5.4 Release Notes*.

AUTOGEN sets the values of system parameters and the sizes of the page and swap files according to the system's workload. Digital recommends that you run AUTOGEN from SAVPARAMS through TESTFILES on a weekly basis and examine AGEN$PARAMS.REPORT to determine the need for additional changes.

Hard coded values in MODPARAMS.DAT affect AUTOGEN's calculations of the feedback parameters. For AUTOGEN to properly calculate values, you should replace the hard coded values in MODPARAMS.DAT with MIN_ values.

For information on using AUTOGEN, see the *Guide to Setting Up a VMS System*.

After you run AUTOGEN, use the checklist in Section 4.1 to make sure you have completed all the tasks described in this chapter.

Part III

Part III describes how to upgrade to Version 5.4 of the VMS operating system.

5 Overview of the Upgrade Procedure

Use the upgrade procedure when you want to obtain a higher version of the VMS operating system. The upgrade procedure does not initialize the system disk (as does the installation procedure). This means you can obtain the latest version of the operating system while keeping your user files and optional software intact.

The Version 5.4 upgrade procedure gives you the opportunity to install VMS DECwindows software on the system disk.

Note: The Version 5.4 upgrade procedure provides new files and directories under [VMS$COMMON...]. If you had any special protections and ACLs before the upgrade, you need to re-apply them to re-establish the security environment you previously had. For more information on creating and maintaining a secure environment, see the *Guide to VMS System Security*.

The Version 5.4 upgrade procedure leaves your VMS license intact. You do not need to re-install the VMS license after you upgrade. (See the *VMS Version 5.4 Release Notes* for detailed information on license management for VMS Version 5.4.)

Before you can upgrade to VMS Version 5.4, your system must be running at least Version 5.0 of the VMS operating system. For example, you can upgrade to Version 5.4 of the VMS operating system from any of the following versions (or related hardware releases):

VMS Version 5.0
VMS Version 5.0-1
VMS Version 5.0-2
VMS Version 5.0-2A
VMS Version 5.1
VMS Version 5.1-B
VMS Version 5.1-1
VMS Version 5.2
VMS Version 5.2-1
VMS Version 5.3
VMS Version 5.3-1
VMS Version 5.3-2

If your system is running an earlier VMS version, such as Version 4.7, you must upgrade to Version 5.0 of the VMS operating system first. For a Version 5.4 rolling upgrade, all systems in the VAXcluster environment must be running at least Version 5.3 of the VMS operating system.

To perform a successful upgrade, you must understand the basic operations of the VAX computer that you are upgrading. Refer to the VMS upgrade and installation supplement for your VAX computer. If you have a VAXcluster environment, refer to the *VMS VAXcluster Manual*.

Overview of the Upgrade Procedure

This chapter describes what happens during the upgrade procedure. It also tells you what you need and lists some restrictions.

5.1 What Happens During an Upgrade

When you upgrade the VMS operating system, the upgrade procedure does the following:

1. Makes room for the upgrade by purging and deleting select VMS and VMS DECwindows files, but leaves all the user files intact
2. Transfers the VMS and VMS DECwindows files from the distribution media to the system disk

Note: The upgrade puts the most recent versions of the VMS files in SYS$COMMON.

3. Merges the old and the new VMS and VMS DECwindows files
4. Cleans up files and structures used only during the upgrade

5.2 Cautions and Restrictions

Be aware of the following cautions and restrictions before you begin the upgrade:

- The VMS Version 5.4 rolling upgrade is not compatible with VMS Version 5.3. If you are doing a rolling upgrade, and any system in the cluster (other than the one you are upgrading) is running VMS Version 5.3, you must perform some extra steps and reboot that system before applying the upgrade. You do not need to perform these extra steps on the system disk you are upgrading, or on systems running VMS Version 5.3-1 or 5.3-2. For more information see Section 7.2.1.
- The upgrade procedure does *not* work across the network.
- If you have changed the names of system directories on your system disk, the upgrade procedure will not work correctly. Restore your system disk to a standard directory structure before you attempt an upgrade.
- You must save AUTOGEN feedback information before rebooting the system with minimum startup to begin the upgrade. For more information, see Section 6.3.
- If you elect not to install optional VMS or VMS DECwindows software, the upgrade procedure removes existing files for those components from the system disk. You can install optional VMS or VMS DECwindows software during the initial installation, or later using VMSTAILOR and DECW$TAILOR.

Caution: Do not attempt to use VMSINSTAL with OPTIONS G to transfer VMS and VMS DECwindows save sets to another media type. If you upgrade using a kit made in this manner, you will produce an unusable system.

- If you must move the VMS save sets from the distribution media to another location before beginning the upgrade, the save sets must be in the [000000] directory of the new media, or the upgrade will fail.
- If you have a shared system dump file in SYS$COMMON:[SYSEXE], the dump file cannot have the name SYSDUMP.DMP. If you have a shared system dump file with this name, Digital recommends that you rename it to SYS$COMMON:[SYSEXE]SYSDUMP-COMMON.DMP. For more information, see the System Management section of the *VMS Version 5.4 Release Notes*.
- You must not move the system disk or the VMS distribution kit from one drive to another during the upgrade.
- You cannot upgrade a shadowed system disk. If your system disk is part of a shadow set, you must create a non-shadowed system disk before beginning the upgrade. For more information, see the *VMS Volume Shadowing Manual*.
- The upgrade procedure deletes all the operator and accounting logs. To save these files, move them to a user directory before you start the upgrade.
- Before you start the upgrade procedure in a VAXcluster environment, you must shut down all the nodes that boot from the system disk you are upgrading, including the node you perform the upgrade on.
- You should record the upgrade procedure on either a hardcopy terminal or a printer attached to the console terminal. If you do not do this, the screen messages will be lost. You will need a transcript in case there is a problem. For information on how to record the procedure, see the VMS upgrade and installation supplement for your VAX computer.
- The upgrade procedure is designed so that you should not have to re-install most layered products after the upgrade. However, you might need to re-install certain layered products because of product-specific installation procedures.

  For example, you must re-install layered products that either create directories synonymous with system directories or use VMS-defined data structures. If a layered product is available (refer to Appendix D), yet exhibits unexpected behavior once Version 5.4 of the VMS operating system is running, check the *VMS Version 5.4 Release Notes* for layered product restrictions. If problems persist, contact your Digital representative.

5.3 Items You Need for the Upgrade

You need the following items to perform the upgrade to Version 5.4 of the VMS operating system:

- VMS Version 5.4 software distribution kit

- Scratch console media (media that does not contain files you want to save) for the following VAX computers:

  VAX 8600, 8650
  VAX 8200, 8250, 8300, 8350
  VAX–11/730
  VAX–11/750
  VAX–11/780, 785

  For example, if you have a VAX–11/780 or VAX–11/785, you need a scratch RX01 floppy diskette.

To prepare for an upgrade, follow the instructions in Chapter 6.

6 Before You Do an Upgrade

Before you do an upgrade, you need to perform several important tasks to prepare the system. This chapter provides step-by-step instructions to perform each of these tasks.

6.1 Pre-Upgrade Checklist

If you are an experienced system manager and are already familiar with these tasks, use the following checklist to make sure you have performed all the tasks before beginning the upgrade. System Managers with less experience are advised to follow the step-by-step instructions beginning at Section 6.2 and to use the following checklist after they have completed all the instructions in this chapter.

- ☐ Read the *VMS Version 5.4 New Features Manual* and the *VMS Version 5.4 Release Notes*.
- ☐ Read the cautions and restrictions described in Section 5.2.
- ☐ Set up your system to record the upgrade procedure on either a hardcopy terminal or a printer attached to the console terminal. If you do not do this, the screen messages will be lost. You will need a transcript in case there is a problem. For information on how to record the procedure, see the VMS upgrade and installation supplement for your VAX computer.
- ☐ Check your AUTOGEN feedback information to be sure it is valid before rebooting the system (described in Section 6.3).
- ☐ Make a backup copy of the system disk and, if applicable, the console media (described in the upgrade and installation supplement for your computer).
- ☐ If, in previous versions, your distribution kit provided standalone BACKUP on one of the following media types, make sure you have created a standalone BACKUP kit (described in Section 6.4):

  RX33
  TU58
  RX50
  RX01
  RL02

- ☐ Make sure SYSUAF.DAT is in SYS$SYSTEM (described in Section 6.5).
- ☐ Make sure PAGEFILE.SYS and SWAPFILE.SYS (if it is located on the system disk) are in SYS$SPECIFIC:[SYSEXE] (described in Section 6.5).
- ☐ Make sure the page file is at least 4600 blocks (described in Section 6.5).

Before You Do an Upgrade

6.1 Pre-Upgrade Checklist

- ☐ If you have a shared system dump file in SYS$COMMON:[SYSEXE], make sure the file is not named SYSDUMP.DMP (described in Section 6.5).
- ☐ Make sure the system disk has at least 36,000 free blocks (described in Section 6.5).
- ☐ If your system disk is part of a shadow set, create a non-shadowed system disk to upgrade (described in the *VMS Volume Shadowing Manual*).
- ☐ Set STARTUP_P1 to MIN (described in Section 6.5).
- ☐ If your system has a KFQSA controller, make sure the VMS5 parameter is set to 0 (described in Section 6.5).
- ☐ Set up the system to automatically restart (described in the VMS upgrade and installation supplement for your computer).
- ☐ Shut down and reboot the system.
- ☐ Set interactive logins to 0.
- ☐ Shut down DECnet–VAX (described in Section 6.6).
- ☐ Stop the queue manager (described in Section 6.6).
- ☐ Make sure you have the following information (described in Section 6.7) readily available:
  - The device names of the drives that will hold the system disk and the distribution media.
  - Passwords of at least 8 characters for the SYSTEM, SYSTEST, and FIELD accounts.
  - A list of optional VMS components you wish to install.
  - A list of optional DECwindows components you wish to install.
  - The information listed on Product Authorization Keys (PAKs) for any new product licenses you will register.

To upgrade a standalone system, go to Chapter 8.

To upgrade a VAXcluster environment, go to Chapter 7.

6.2 Understanding Version 5.4 of the VMS Operating System

Before you begin the upgrade, read the *VMS Version 5.4 New Features Manual* and the *VMS Version 5.4 Release Notes*. These manuals describe the changes made to the VMS operating system in Version 5.4.

6.3 Checking AUTOGEN Feedback Data

After the upgrade is finished, AUTOGEN runs. If it finds valid feedback data on your system, it tries to use AUTOGEN feedback. Feedback data is produced by the SAVPARAMS phase of AUTOGEN and is stored in SYS$SYSTEM:AGEN$FEEDBACK.DAT. Valid feedback data is defined as follows:

- The system was up for at least 24 hours when the feedback data was gathered
- The feedback data is not over 30 days old

Note: If your feedback data is NOT valid, AUTOGEN will run WITHOUT feedback (all previous feedback-generated parameters will be replaced by the AUTOGEN NOFEEDBACK calculations).

To insure that AUTOGEN feedback will run after the upgrade, do the following:

1. Check the validity of your current feedback data by entering the following command:

   ```
   $ @SYS$UPDATE:AUTOGEN GETDATA TESTFILES
   ```

   This command is for test purposes and does not result in the modification of SYSGEN parameters or changes to the page and swap files. If your current feedback is valid, AUTOGEN will use it during the upgrade.

   AUTOGEN will issue several warnings if your current feedback data is invalid. If your current feedback data is invalid, go to step 2.

2. If your current feedback is invalid, you can generate new feedback data (assuming your system was up for at least 24 hours). Enter the following command:

   ```
   $ @SYS$UPDATE:AUTOGEN SAVPARAMS TESTFILES
   ```

   This command is for test purposes and does not result in the modification of SYSGEN parameters or changes to the page and swap files. Again, AUTOGEN warns you if the feedback data just produced is invalid.

3. If you follow steps 1 or 2 and your current feedback data is valid, AUTOGEN uses your current feedback generated parameters during the upgrade. Your current feedback generated parameters will not be lost during the upgrade.

6.4 Backup Tasks

Digital recommends that you make a backup copy of the system disk and, if your configuration allows it, upgrade the *backup copy*. If there are problems, you will still have a working system disk. To make a backup copy of the system disk, follow the instructions in the VMS upgrade and installation supplement for your VAX computer.

If you have a VAXcluster environment with more than one system disk, make sure the volume label on each system disk and backup copies of system disks are unique. Use the SET VOLUME/LABEL command to change a volume label, if necessary.

If you have one of the following VAX computers, you also must make a backup copy of the console media:

VAX–11/730
VAX–11/750
VAX–11/780, 785
VAX 8200, 8250, 8300, 8350
VAX 8600, 8650

The upgrade procedure updates the console media for you. Use the backup copy of the console media during the upgrade procedure. If there are problems, you will still have a working copy of the console media. To make a backup copy of the console media, use the CONSCOPY.COM procedure as described in the VMS upgrade and installation supplement for your VAX computer.

Caution: If your VAX computer uses one of the following console media, you did not receive a new standalone BACKUP kit with your VMS distribution kit:

TU58 tape cartridge
RX33 diskette
RX50 diskette
RX01 diskette
RL02 disk cartridge

Before continuing with the upgrade, make sure you have a standalone BACKUP kit. If you do not have one, create a standalone BACKUP kit using the STABACKIT.COM procedure as described in the VMS upgrade and installation supplement for your VAX computer. You can create standalone BACKUP on console media or on your system disk.

6.5 Preparing the System Disk

Prepare the system disk for the upgrade using the following procedure:

1. Log in to the SYSTEM account.
2. If you are using a volume-shadowed system disk, you must create a non-shadowed system disk to upgrade. Follow the instructions in the *VMS Volume Shadowing Manual* to prepare a shadowed system disk for the upgrade.
3. If you need to boot the backup copy of your system disk, or if you are upgrading a VAXcluster environment, enter the following command (otherwise, go to step 5):

   ```
   $ @SYS$SYSTEM:SHUTDOWN
   ```

   When the procedure asks if an automatic system reboot should be performed, press Return for NO. If you are a VAXcluster member, choose the CLUSTER_SHUTDOWN option. When the procedure is finished, halt the system as described in the VMS upgrade and installation supplement for your VAX computer.

4. If applicable, remove the system disk and replace it with the backup copy.
5. Boot the system from the backup copy of the system disk. If you do not know how to boot the system, refer to the VMS upgrade and installation supplement for your VAX computer.
6. Log in to the SYSTEM account.
7. The current copy of SYSUAF.DAT must be on the system disk in SYS$SYSTEM. Otherwise, you cannot proceed with the upgrade. If you have moved your copy of SYSUAF.DAT to another location, copy it to SYS$SYSTEM for the duration of the upgrade.

Note: If you are doing a rolling upgrade, the shared version of SYSUAF.DAT is not updated.

Move any files that you do not want changed into a user directory.

8. If you have a shared system dump file in SYS$COMMON:[SYSEXE], the dump file cannot have the name SYSDUMP.DMP. If you have a shared system dump file with this name, Digital recommends that you rename it to SYS$COMMON:[SYSEXE]SYSDUMP-COMMON.DMP. For more information, see the *VMS Version 5.4 Release Notes*.
9. To analyze the system disk for inconsistencies and errors in the file structure, enter the following command:

```
$ ANALYZE/DISK_STRUCTURE SYS$SYSDEVICE
```

Ignore the following message:

```
%ANALDISK-I-OPENQUOTA, error opening QUOTA.SYS
```

If you find any other errors on the system disk, repair the errors by entering the following command:

```
$ ANALYZE/DISK_STRUCTURE/REPAIR SYS$SYSDEVICE
```

10. Make sure the page file (PAGEFILE.SYS) and, if appropriate, the swap file (SWAPFILE.SYS) are in SYS$SPECIFIC:[SYSEXE].

    The upgrade requires that the page file be located on the system disk in SYS$SPECIFIC:[SYSEXE]. The swapfile (SWAPFILE.SYS) does not need to be located on the system disk for the upgrade, but if it is located on the system disk, the file must be located in SYS$SPECIFIC:[SYSEXE].

    If the upgrade procedure finds these files in the common directory, it will move them to the specific directory before continuing.

11. Check the page file to make sure it is at least 4600 blocks. Use the following procedure:

    a. To find out the size of the page file, enter the following command:

```
$ @SYS$UPDATE:SWAPFILES
```

    The procedure displays the size of the page file and asks you to enter a new value, as follows:

```
Enter new size for paging file:
```

If the page file (PAGEFILE.SYS) size is greater than 4600 blocks, press the Return key. If it is less than 4600 blocks, enter 4600 and press Return.

b. If the swap and dump files are on the system disk, the procedure displays the sizes of each file. To keep the current values, press Return after each prompt.

If the swap file is not on the system disk, the procedure asks if you want to create one. You do not need swap and dump files on the system disk for the upgrade procedure. Enter the appropriate response for your system.

If you changed the size of the page file or created a swap file, a message is displayed that tells you to reboot the system. Do not reboot now. You will reboot later.

12 Check the system disk to make sure there is enough room for the upgrade. To determine whether you have enough room, enter the following command:

```
$  SHOW DEVICE SYS$SYSDEVICE
```

To upgrade the VMS *required* files, the number of free blocks must be at least 36,000. If you plan to upgrade the VMS *library* or *optional* files, or VMS DECwindows, you will require more free blocks. The following table shows the number of free blocks required to upgrade each VMS and DECwindows component:

Component	Free Blocks Required
VMS Library files	30,000
VMS Optional files	13,000
DECwindows device support	13,500
DECwindows applications	12,000
DECwindows programming support	26,500
DECwindows examples	3,000

13 During the upgrade procedure you will shut down and reboot the system a number of times. To avoid startup of certain system products and layered products on the system during each reboot, set the SYSGEN parameter STARTUP_P1 to MIN. Enter the following commands, pressing the Return key after each one:

```
$  RUN SYS$SYSTEM:SYSGEN
SYSGEN>  USE CURRENT
SYSGEN>  SET STARTUP_P1 "MIN"
SYSGEN>  WRITE CURRENT
SYSGEN>  EXIT
```

14 Make sure the SYSGEN parameter VMS5 is set to 0. If your system has a KFQSA controller, the VMS5 parameter may have been set to a non-zero value for a previous upgrade. The VMS5 parameter is reserved for Digital use only.

To check the value of the VMS5 parameter, enter the following commands, pressing the Return key after each one:

```
$  RUN SYS$SYSTEM:SYSGEN
SYSGEN>  USE CURRENT
SYSGEN>   SHOW VMS5
Parameter Name              Current   Default   Minimum   Maximum Unit  Dynamic
--------------              -------   -------   -------   ------- ----  -------
VMS5                              1         0         0        -1
```

If the current value of VMS5 is not 0, set it to 0 by entering the following commands:

```
SYSGEN>  SET VMS5 0
SYSGEN>  WRITE CURRENT
SYSGEN>  EXIT
```

15 Any SYSGEN parameters that you modified and *did not* enter in MODPARAMS.DAT are lost during the upgrade. To retain these parameters, enter their names in MODPARAMS.DAT and the value that AUTOGEN needs to add to the default minimum value.

For example, if you modified GBLPAGES by 128 pages above the default, enter 128 into MODPARAMS.DAT, as follows:

```
ADD_GBLPAGES=128
```

When AUTOGEN runs after the upgrade it uses the values in MODPARAMS.DAT.

16 To prepare the system for the upgrade, go to Section 6.6.

6.6 Preparing the System

To prepare the system for the upgrade, use the following procedure:

1 If your computer uses console media, set the computer to restart automatically. For some VAX computers this means setting one or more switches. For others, this means entering certain console mode commands. To determine how to enable auto restart, refer to the VMS upgrade and installation supplement for your VAX computer.

If you have a system that does not use console media, set your system so you can reboot it manually. For more information, refer to the VMS upgrade and installation supplement for your VAX computer.

2 Shut down and reboot the system to activate any SYSGEN parameters that you changed (STARTUP_P1, for example). To shut down the system, enter the following command:

```
$  @SYS$SYSTEM:SHUTDOWN
```

When the procedure asks if an automatic system reboot should be performed, type Y (for YES) and press the Return key.

3 After the system reboots, log in to the SYSTEM account.

4 To prevent users from logging in to the system during the upgrade, enter the following command:

```
$  SET LOGINS/INTERACTIVE=0
```

5 To determine whether your system is running DECnet–VAX software, enter the following command:

```
$  SHOW NETWORK
```

If the system displays the following message, go to step 6:

```
%SHOW-I-NONET, network unavailable
```

If the system displays information on the status of the VAX/VMS network, enter the following commands to shut down the network:

```
$  RUN SYS$SYSTEM:NCP
NCP>  SET EXECUTOR STATE OFF
NCP>  EXIT
```

6 To stop all batch and print queues, enter the following command:

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

6.7 Information You Need During the Upgrade

At different points during the upgrade, you will be prompted to supply information. Make sure you have this information readily available. The information you will need includes the following:

- Device names of the drives that will hold the system disk and the distribution media.
- Passwords of at least 8 characters for the SYSTEM, SYSTEST, and FIELD accounts. You will be prompted to enter passwords of at least 8 characters for each of these accounts.
- A list of optional VMS components you wish to install. If you want to save space on your system disk, you can choose to not install the *library* and *optional* VMS components. The installation procedure prompts you to specify whether you want to install each of these components. Appendix B lists the files included in these components.
- A list of the DECwindows components you want to install. If you want to save space on your system disk, you can choose to *not* install certain DECwindows components. The installation procedure prompts you to specify which of the following DECwindows components you want to install: device support, applications, programming support, example files. Appendix C lists the files included in each of these components.
- The information listed on Product Authorization Keys (PAKs) for any new product licenses you will register. You do not need to re-register your VMS license, or the licenses for any system-integrated products you are already running. If you wish to register new licenses, however, the upgrade procedure gives you the opportunity to do so. You can also register them after the upgrade is complete using VMSLICENSE.COM. To register a license, you need to enter the information listed on the accompanying PAK.

Once you have completed these steps, use the checklist in Section 6.1 to make sure you have completed all the necessary tasks before continuing with the upgrade.

7 Upgrading a VAXcluster Environment

This chapter describes how to upgrade a VAXcluster environment. Refer to the *VMS VAXcluster Manual* for more detailed information on VAXcluster environments.

There are two types of cluster upgrades: concurrent and rolling. The type of upgrade you use depends on whether you want to maintain the availability of the cluster during the upgrade and whether you have more than one system disk:

- Concurrent upgrade—During a concurrent upgrade, you must shut down the entire cluster and upgrade each system disk. No one can use the cluster until you upgrade each system disk, and reboot each VAX computer. If all systems in the VAXcluster environment boot from one system disk, you must perform a concurrent upgrade. To perform a concurrent upgrade, see Section 7.1.
- Rolling upgrade—During a rolling upgrade, you can keep some of the VAX computers in the cluster running while you upgrade others (assuming that you have more than one system disk). The cluster runs temporarily with different versions of the VMS operating system. To perform a rolling upgrade, see Section 7.2.

7.1 Performing a Concurrent Upgrade

You perform a concurrent upgrade by shutting down the entire cluster, upgrading each system disk, and rebooting each VAX computer in the cluster. When the cluster reboots, each VAX computer will be running the upgraded version of the VMS operating system. Users cannot access the cluster until you upgrade each system disk and reboot each VAX computer.

To perform a concurrent upgrade, use the following procedure:

1. Shut down the entire cluster by logging in to the SYSTEM account on each node *locally*, entering the following command, and choosing the CLUSTER_SHUTDOWN option:

   ```
   $ @SYS$SYSTEM:SHUTDOWN
   ```

2. If you have only one system disk for your cluster, go to step 3.

   If you have more than one system disk, select one to upgrade.

3. Perform a conversational boot of a single VAX computer from the system disk you will be upgrading. For more information about the conversational boot procedure, refer to the VMS upgrade and installation supplement for your VAX computer.

At the SYSBOOT> prompt, use the SHOW VOTES and SHOW EXPECTED_VOTES command to display the current values of the parameters. Write these values down because you will need to restore them later. Set the VOTES and EXPECTED_VOTES values to 1, by entering the following commands. Press the Return key after each one:

```
SYSBOOT>  USE CURRENT
SYSBOOT>  SET VOTES 1
SYSBOOT>  SET EXPECTED_VOTES 1
SYSBOOT>  CONTINUE
```

4. Upgrade the system disk, using the procedure in Chapter 8.
5. If you have only one system disk, go to step 6.

   If you have more than one system disk, shut down the system that you just upgraded. Then repeat steps 2 through 4 to upgrade each system disk in the cluster. When you finish upgrading all of the system disks, go to step 6.
6. Reboot each computer in the VAXcluster environment.
7. Complete the post-upgrade procedure described in Chapter 9.

7.2 Rolling Upgrade Overview

During a rolling upgrade, you upgrade each system disk individually, allowing old and new versions of the VMS operating system to run together temporarily in the same cluster. Clusters running two versions of the VMS operating system are called mixed-version clusters. Because rolling upgrades allow mixed-version clusters, you maintain availability of the systems that you are not upgrading.

CAUTION: To perform a Version 5.4 rolling upgrade, the other nodes in the cluster must be running at least Version 5.3 of the VMS operating system.

However, if any other system in the cluster is running Version 5.3, you must perform extra steps and reboot each Version 5.3 system when performing the rolling upgrade. These steps are described in Section 7.2.1. You can perform these steps before you begin the upgrade, or while you are performing the upgrade.

You do not need to perform these steps on the system disk you are upgrading, or on systems running VMS Version 5.3-1 or 5.3-2.

Digital recommends that all VAX computers in a cluster run the same version of the VMS operating system. Mixed-version clusters are supported *only* for the purpose of upgrading the entire cluster incrementally.

A rolling upgrade does not apply when all systems boot from a single system disk. When all systems boot from a single system disk, perform a concurrent upgrade using the procedure described in Section 7.1.

7.2.1 Performing a Rolling Upgrade

This section contains step-by-step instructions for performing a rolling upgrade.

7.2.1.1 Special Steps Required for Version 5.3 Systems

If any system disk in the cluster (other than the one you are upgrading) is running Version 5.3 (not 5.3-1 or 5.3-2), you must perform some special steps and reboot these systems to make them compatible with VMS Version 5.4 before the upgrade can be performed. The upgrade procedure described in Section 7.2.1.2 contains instructions for performing these steps.

You might want to perform these steps before attempting the VMS Version 5.4 upgrade. To obtain the files and the instructions for applying the fix, follow these steps (otherwise, go to Section 7.2.1.2):

1. Place the first volume of the VMS kit in the source drive and mount the volume by entering the following command, where ddcu is the device name of the source drive:

```
$   MOUNT/FOREIGN ddcu:
```

2. Enter the following command:

```
$  BACKUP/SELECT=(VMS$ROLLING_UPGRADE_054.TXT,VMS$RU_FIXUP_054.COM, MSCP.EXE_V53-1) -
$_  ddcu:[000000]VMS054.A/SAVE  SYS$COMMON:[SYSUPD]*.*;/LOG
```

   This command will move the required files and the file containing instructions to the [SYSUPD] directory on your system disk.

3. Follow the instructions in the file SYS$COMMON:[SYSUPD]VMS$ROLLING_UPGRADE_054.TXT.

   To display the instructions on your screen, enter the following command:

```
$  TYPE SYS$UPDATE:VMS$ROLLING_UPGRADE_054.TXT
```

   To print the instructions, enter the following command:

```
$  PRINT SYS$UPDATE:VMS$ROLLING_UPGRADE_054.TXT
```

4. Perform the upgrade following the instructions in Section 7.2.1.2. When the procedure displays the following prompt, enter Y (for yes):

```
* Have you applied the required fix to all VMS V5.3 system disks? (Y/N):
```

7.2.1.2 Rolling Upgrade Instructions

To perform a rolling upgrade, use the following procedure:

1. Check the votes and make adjustments to maintain the proper quorum so that the cluster can continue to operate throughout the upgrade. (The *VMS VAXcluster Manual* describes this procedure in detail.)

2. Select a system disk to upgrade.

3. If proper quorum is not maintained at any time during the upgrade procedure, the shutdown procedure will hang the cluster. If the cluster hangs during a shutdown, enter the following commands and press the Return key after each one:

```
$  Ctrl/P
>>>  H
>>>  D/I 14 C
>>>  C
IPC>  Q
IPC>  Ctrl/Z
```

4. Shut down all VAX computers booting from the system disk that you are upgrading, except for the computer that you will use to apply the upgrade.

Note: VAX computers that do not boot from the system disk that you are upgrading remain available to users.

a. Log in to the SYSTEM account on the VAX computer you are shutting down.

b. Enter the following command and press the Return key:

```
$  @SYS$SYSTEM:SHUTDOWN.COM
```

c. On one of the remaining VAX computers in the cluster, enter the following command and press Return:

```
$ SET CLUSTER/EXPECTED_VOTES
```

d. Repeat steps a through c for each VAX computer requiring a shutdown.

This procedure lets one VAX computer continue running from the system disk (assuming computers running from different system disks supply enough votes to sustain cluster quorum).

5. Upgrade the system disk, using the procedure in Chapter 8.

CAUTION: When you upgrade the system disk, be sure to use a local drive for the distribution media (rather than a drive that is accessible to the cluster). The upgrade procedure requires exclusive access to the drive that holds the distribution media.

6. During phase 1 of the upgrade, the procedure will ask if you are performing a rolling upgrade. If you enter Y, the procedure asks you several questions to determine if any system disk in the cluster (other than the one you are upgrading) is running VMS Version 5.3. For example:

```
* Are the other system disks running VMS V5.3 or higher?
```

All systems in the cluster must be running at least VMS version 5.3 to perform the rolling upgrade. Type Y and press the Return key. The procedure displays the following message:

```
    If you are running VMS version 5.3 (not the same as 5.3-n) on
    any other system disks while doing this upgrade, a rolling
    upgrade can not be performed until a fix has been applied.

* Is VMS V5.3 running on any other system disks?
```

This question does not apply to the system disk you are upgrading. If the other system disks are running VMS Versions 5.3-1, 5.3-2, or both, enter N (for no), and proceed with the upgrade. Go to step 7 in this list.

If any other system disk in the cluster (other than the one you are upgrading) is running VMS Version 5.3, type Y and press Return. The procedure displays the following message:

```
* Have you applied the required fix to all VMS V5.3 system disks? (Y/N):
```

If you have already performed the steps described in Section 7.2.1.1, enter Y, continue with the upgrade, and go to step 7.

Otherwise, enter N. You must apply a required fix and reboot each VMS Version 5.3 system. To do so, follow these steps:

a. The procedure displays the following message:

```
*****************************  NOTICE *****************************

This rolling upgrade is not compatible with the 5.3 system.
In order to perform a rolling upgrade with 5.3, you must take
the following action:

1. Read SYS$UPDATE:VMS$ROLLING_UPGRADE_054.TXT on this system disk

2. Perform documented steps

3. Restart upgrade procedures

*****************************************************************
```

b. To read SYS$UPDATE:VMS$ROLLING_UPGRADE_054.TXT, type the following command and press Return:

```
$ TYPE SYS$UPDATE:VMS$ROLLING_UPGRADE_054.TXT
```

To print this file, type the following command and press Return:

```
$ PRINT SYS$UPDATE:VMS$ROLLING_UPGRADE_054.TXT
```

c. Perform the steps described in the file.

d. Restart the upgrade procedure following the instructions in Chapter 8. When the procedure prompts you to specify if you have applied the required fix, enter Y and continue with the procedure.

7 When the upgrade is complete, reboot each VAX computer that boots from the system disk you have just upgraded.

8 Repeat steps 2 through 7 for each system disk you are upgrading. Once you have performed steps **a** through **d** in step 6, you do not need to perform them again.

9 Perform the post-upgrade tasks described in Chapter 9.

8 Standalone Upgrade

This chapter explains the upgrade procedure for a single VMS system. Before you begin the upgrade procedure, read the overview of the upgrade procedure described in Chapter 5 and complete the pre-upgrade tasks described in Chapter 6. If you are upgrading a VAXcluster, you must also read Chapter 7.

CAUTION: You cannot upgrade to Version 5.4 of the VMS operating system on the following VAX computers and system disks:

- **MicroVAX I, VAXstation I, VAXstation 8000, and VAX 11/725 computers**
- **RK07, RL02, RC25, RD32, RD51, and RD52 system disks**

The upgrade procedure is divided into six phases. You must complete each phase of the upgrade procedure. On most VAX computers the upgrade procedure continues automatically from phase to phase. Read the screen displays carefully and, if you need to reboot your VAX computer manually, refer to the VMS upgrade and installation supplement for your VAX computer.

8.1 Preparing to Boot from [SYSF]

During the upgrade procedure, you are asked to boot the system from the [SYSF] directory on the system disk. If you do not know how to boot from [SYSF], read the VMS upgrade and installation supplement for your VAX computer.

Some VAX computers require a special command procedure to boot from [SYSF]. If you are performing an upgrade on one of the following VAX computers you must modify the boot command procedures, DEFBOO and *ddd*GEN before beginning the upgrade procedure:

VAX 8530, 8550, 8700, 8810, 8820-N
VAX 8820, 8830, 8840

To modify the boot command procedures, exit from the VMS operating system (but do not shut it down) and enter the appropriate console mode. Make copies of DEFBOO and *ddd*GEN before you modify them. Name the copies DEFBOO.SAV and *ddd*GEN.SAV.

The VMS upgrade and installation supplement for your VAX computer describes the procedures for copying and modifying DEFBOO and *ddd*GEN. Use these procedures to modify DEFBOO and *ddd*GEN so that they specify [SYSF] as the directory from which to boot.

In DEFBOO, change the DEPOSIT command, as follows:

```
DEPOSIT R5=F0000000
```

Standalone Upgrade

8.1 Preparing to Boot from [SYSF]

In *ddd*GEN, change the DEPOSIT command that sets boot flags in register 5 (R5), as follows:

```
DEPOSIT R5 F0000001
```

If you are upgrading a VAX 8200, 8250, 8300, or 8350 computer, you must create a boot command procedure that allows you to boot from the [SYSF] directory before beginning the upgrade. The VMS upgrade and installation supplement for your VAX computer describes how to create this boot command procedure.

8.2 Configuring Devices

Because you rebooted your system with minimum startup, the system did not automatically configure all devices. However, you might need access to some of the devices to perform the upgrade. Run SYSGEN to reconfigure the devices on the system and run STARTUP CONFIGURE by entering the following commands:

```
$  RUN SYS$SYSTEM:SYSGEN
SYSGEN>  AUTOCONFIGURE ALL
SYSGEN>  EXIT
$  @SYS$SYSTEM:STARTUP CONFIGURE
```

8.3 Beginning the Upgrade Procedure

This section describes how to start the upgrade procedure using VMSINSTAL. You can type a question mark (?) for help at any time while running VMSINSTAL.

The instructions refer to the *source-drive* and the *target-drive* where:

- The *source-drive* is the drive that holds the distribution kit.
- The *target-drive* is the drive that holds the system disk.

1 Put the distribution disk or tape in the source-drive. If your VMS distribution kit contains several disks or tapes, put the first one in the drive. The paper labels on the disks or tapes indicate the order in which you use them.

2 To start VMSINSTAL, use the following procedure:

a. Log in to the SYSTEM account.

b. Type the following command and press the Return key:

```
$  @SYS$UPDATE:VMSINSTAL
```

VMSINSTAL displays messages similar to the following:

```
        VAX/VMS Software Product Installation Procedure V5.3

It is 19-JUN-1990 at 15:00.00.00
Enter a question mark (?) at any time for help.
```

3 The procedure displays the following message:

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

If you have made a backup of the system disk, press Return and go to step 4.

If you have not made a backup of the system disk, do the following:

a. Type N (for NO) and press Return. VMSINSTAL ends and the dollar-sign prompt ($) is displayed.

b. Back up the system disk as described in the VMS upgrade and installation supplement for your VAX computer.

c. After you have made a backup of the system disk, start the upgrade procedure again from the beginning of Section 8.3.

4 VMSINSTAL asks for the name of the drive that holds the distribution media:

```
* Where will the distribution volumes be mounted:
```

Enter the device name of the source drive. For example, if you have a VAX 6000 series computer that has a TK70 with a device name of MUC6, enter the following command and press the Return key:

```
* Where will the distribution volumes be mounted:  MUC6
```

If the drive is connected to an HSC, enter the HSC name and the device name of the drive separated by a dollar sign ($). For example, if the HSC name is MUTT, and the device name is DJA2, enter the following and press Return:

```
* Where will the distribution volumes be mounted:  MUTT$DJA2
```

If VMSINSTAL displays an error message, make sure the drive is connected and set up properly. Enter the name again. If VMSINSTAL displays another error message, verify the device name using the following procedure:

a. Press Ctrl/Y.

b. Enter the SHOW DEVICE command to make sure you have the correct device name.

c. Start the upgrade procedure again from the beginning of Section 8.3.

5 When VMSINSTAL displays the following message, type VMS054 and press the Return key. Make sure that you use the number zero when you type VMS054. For example:

```
* Products:  VMS054
```

6 When the procedure displays the following message, press Return:

```
* Enter installation options you wish to use (none):
```

7 VMSINSTAL displays a message similar to the following:

```
Please mount the first volume of the set on _DJA2.
```

When the procedure displays the following message, type Y (for YES) and press Return:

```
* Are you ready?   Y
```

The procedure displays messages similar to the following:

```
%MOUNT-I-MOUNTED, VMS054 mounted on _DJA2:
The following products will be processed:
  VMS V5.4
```

After a slight pause, the procedure displays messages similar to the following:

```
        Beginning installation of VMS V5.4 at 15:05
%VMSINSTAL-I-RESTORE, Restoring product saveset A...
                          VAX/VMS V5.4 Upgrade Procedure
                   Model: VAX 6000-410
            System device: RA82 - KROK$DUA1:
              System root: [SYS0]
              Free blocks: 752667
                 CPU type: 11-02
```

The upgrade procedures display several messages that provide you with the following information:

— What VMSINSTAL is doing.

— Notes, suggestions, and restrictions about various parts of the upgrade.

— Status of the upgrade.

Read these messages carefully to decide whether or not you need to interrupt the upgrade procedure.

8 An interruption is allowed before Phase 1. The procedure displays the following message:

```
* Do you want to continue? (Y/N):
```

To continue the upgrade, type Y and press the Return key.

To interrupt the upgrade, do the following:

a. Type N and press Return.

b. If prompted for another product, press CTRL/Z to exit from VMSINSTAL.

9 The procedure displays the current date and time. If the date and time are correct, type Y. If they are incorrect, type N. Enter the correct date and time using the 24-hour clock format and press Return. For example:

```
PLEASE ENTER DATE AND TIME (DD-MMM-YYYY HH:MM)  19-JUN-1990 15:10
```

10 Continue with Phase 1 of the upgrade.

8.4 Upgrade Phase 1

This section describes Phase 1 of the upgrade. During this phase you provide passwords for system accounts and select VMS and VMS DECwindows options and the procedure restores the files in the VMS *required* save set.

At the end of this phase, the procedure shuts down your VAX computer. If you have a computer with console media, the system will probably automatically reboot. If your computer does not use console media, you must manually reboot from the SYSF directory.

At the beginning of Phase 1, the procedure displays a message similar to the following:

```
        Upgrade Phase 1  19-JUN-1990 15:11:17.05

* Are you performing a rolling upgrade? (Y/N):
```

1 If you are performing a concurrent upgrade or an upgrade on a standalone system, type N and press the Return key.

If you are performing a rolling upgrade, type Y and press Return. Follow the instructions in Section 7.2.1.

2 To ensure system security, the upgrade procedure requires you to enter passwords for the SYSTEM, SYSTEST, and FIELD accounts before continuing. Passwords must be at least 8 characters in length; they do not appear on the display. Press Return after you enter each one.

After you enter the passwords, the procedure checks each one to make sure it meets the requirements for a good password. For example:

```
To help provide a more secure system, you are asked to enter passwords
for key system accounts. Passwords must be a minimum of 8 characters
in length. You may enter new or existing passwords.

Enter passwords for the following accounts:

             SYSTEM, SYSTEST, FIELD

* Enter password for SYSTEM:
* Re-enter for verification:
%UAF-I-MDFYMSG, user record(s) updated
%VMS-I-PWD_OKAY, account password for SYSTEM verified

* Enter Password for SYSTEST
* Re-enter for verification:
%UAF-I-MDFYMSG, user record(s) updated
%VMS-I-PWD_OKAY, account password for SYSTEST verified

     The SYSTEST_CLIG account will be disabled. You must re-enable
     it before running UETP but do not assign a password.)

Enter new password for account FIELD:
Re-enter for verification:
%UAF-I-MDFYMSG, user record(s) updated
%VMS-I-PWD_OKAY, account password for FIELD verified
```

If you do not get password errors, go to step 3.

If you enter a password incorrectly or if the password is too easy to guess, the procedure displays error messages similar to the following:

```
%VMS-W-PWD_INVALID, account password for FIELD is invalid
-VMS-I-PWD_WEAK, password is too easy to guess

Because of the preceding error, you must take action to secure this account.
You must either disable this account, change its password, or do both.
```

When the procedure asks if you want to disable the account, type N and press the Return key. When the procedure asks if you want to enter a new password, type Y and press Return. Then enter a new password. For example:

```
Do you want to disable the account (Y/N)?  N
Do you want to change the account password (Y/N)?  Y
You must now select a new primary password for the FIELD account. The
password you select must be at least 8 characters in length and may not be the
same as the name of the account.

New password:
Verification:
%UAF-I-MDFMSG, user record(s) updated
%VMS-I-PWD-SET, primary password for account FIELD set
```

3 The procedure does the following:

- Checks the value of the FILLM quota in the DEFAULT, SYSTEM, and various other accounts and modifies them if necessary.
- Turns off quotas on the system disk
- Stops OPCOM
- Stops the error formatter (ERRFMT)

4 If your VAX computer is *not* one of the following, go to step 6:

VAX 11/730
VAX 11/750
VAX 11/780
VAX 8200, 8250, 8300, 8350
VAX 8600, 8650
VAX 9000

If your VAX computer is included in the preceding list, the procedure can temporarily modify the DEFBOO.CMD (or DEFBOO.COM if you have a VAX 8600 computer) to boot from the [SYSF] directory. During Phase 4 of the upgrade, the procedure restores the DEFBOO.CMD (or DEFBOO.COM) file that is set up to boot from the current system disk and system root.

The procedure displays a message similar to the following:

```
* Do you want to boot from the [console media]?
```

If you prefer to boot your system manually from [SYSF], type N, press the Return key, and go to step 5.

If you are booting directly from a local system disk, type N and press Return. If you are using a CI750, type Y and press Return.

If you typed Y, the procedure will ask you to insert the console media in the drive. Use the backup copy of the console media. Insert the backup copy in the drive and do not remove it from the drive until the upgrade has been completed. Make sure the console media is write-enabled.

5 The procedure displays the following message:

```
* Do you want to continue? (Y/N)
```

To interrupt the upgrade, type N and press Return.

To continue the upgrade, type Y and press Return.

6 The procedure does the following:

- Cleans up the system disk by purging all accounting data files, operator logs, and the directory SYSERR. The procedure also deletes all JNL files in the root directory and its subdirectories.
- Builds the directory tree [SYSF] and deletes the VMS and VMS DECwindows files that are not needed to reboot during Phase 1 of the upgrade.

As the procedure performs each of these tasks, it displays messages describing what is being done. When this step is complete, the procedure displays a message similiar to the following:

```
File cleanup complete - 16-JUN-1990 15:29:41.92
```

7 The procedure displays information about VMS and VMS DECwindows files that are optional. The information displayed is similiar to the following:

```
Select optional software you wish to have installed. You can choose to
install one or more of the following VMS and/or DECwindows components:

    o VMS library                    - 30000 blocks
    o VMS optional                   - 13000 blocks
    o DECwindows device support      - 13500 blocks
    o DECwindows applications        - 12000 blocks
    o DECwindows programming support - 26500 blocks
    o DECwindows example files       -  3000 blocks

Space remaining:  251165 blocks
```

You must indicate which options you want to install.

CAUTION: If you elect to not install one or more of the optional VMS or DECwindows components at this time, you can install the files after the upgrade using the VMSTAILOR and DECW$TAILOR utilities. Please note that for TK50 tape cartridges, it is quicker to install VMS and DECwindows software during the upgrade than to tailor on optional components after the upgrade. For information about using VMSTAILOR or DECW$TAILOR, see Chapter 12.

To select the options you want to install, follow the instructions displayed on the screen. For example:

```
* Do you want to install the VMS library files? (Y/N)
```

8 The contents of the VMS *library* save set are listed in Appendix B. If you want to install the VMS *library* files, type Y and press the Return key. If you do *not* want to install the VMS *library* files, type N and press Return.

The procedure displays the following message:

```
* Do you want to install the VMS optional files? (Y/N)
```

9 The contents of the VMS *optional* save set are listed in Appendix B. If you want to install the VMS *optional* files, type Y and press Return. If you do *not* want to install the VMS *optional* files, type N and press Return. The procedure displays the following message:

```
    You can provide DECwindows support now, or you can use the
    DECW$TAILOR utility to provide or remove DECwindows support
    after the upgrade.

    NOTE: Some media, TK50s in particular, can be very slow when
    tailoring on files. You might want to provide DECwindows
    options now and tailor off unwanted files later.

*Do you want to provide optional DECwindows support? (Y/N)
```

10 You must indicate if you wish to install DECwindows software.

If you do not want to install VMS DECwindows software, type N and go to step 11. Note that VMS DECwindows software is useful even on a nonworkstation VAX computer, because you can run VMS DECwindows applications on a nonworkstation VAX computer and display them on a workstation.

If you do want to install VMS DECwindows software, perform the following steps:

a. Type Y and press the Return key. The procedure displays the following message:

```
You can install either the complete DECwindows kit or selected portions
of it. A complete installation requires 56000 blocks. If you
want to perform a partial installation, you can choose to install one
or more of the following DECwindows components:

    o DECwindows device support      -  13500 blocks
    o DECwindows applications        -  12000 blocks
    o DECwindows programming support -  26500 blocks
    o DECwindows example files       -   3000 blocks

* Do you want to install complete DECwindows support? (Y/N)
```

b. To install the complete DECwindows kit, type Y and press Return. The procedure displays the amount of remaining disk space and the DECwindows options that will be provided.

To install selected portions of the VMS DECwindows kit, type N and press Return. The procedure displays information about the amount of space you need on the system disk for each option. As you choose your options, the procedure calculates the amount of space that would be left on the system disk.

Consider the following before you select the VMS DECwindows options that you want to install:

- Device support files—If you are installing VMS DECwindows on a VAXstation or in a cluster that includes VAXstations, you need these files. Otherwise, you do not need them. Please note that if you select the device support files, you automatically get applications and 75 dots-per-inch (75 DPI) video font files, and you will be prompted to indicate if you want to install 100 dots-per-inch (100 DPI) video font files.
- Applications files—VMS DECwindows provides a variety of applications including the following: Bookreader, Calculator, Calendar, Cardfiler, CDA Document Viewer, Clock, EVE Text Editor, FileView, Mail, Notepad Text Editor, Paint Graphics Editor, DECterm, the Session Manager, and the Icon Box. If you are installing VMS DECwindows on a VAXstation, you need these files. They are also useful on a nonworkstation VAX computer.
- 100 dots-per-inch (100 DPI) video font files—If you have a workstation monitor capable of displaying 100 DPI (for example, a VR150, VR160, or VR295), selecting 100 DPI fonts will give you a more readable display when you run VMS DECwindows applications. If your workstation monitor cannot display 100 DPI fonts (for example, a VR260 or VR290), you can save disk space by not selecting this option. For more information on the capabilities of your particular monitor, see the owner's guide for your monitor. You will only be prompted to indicate if you want 100 DPI files if you select the device support option.
- Programming support files—The VMS DECwindows programming support files are needed only to develop applications and are not required to run VMS DECwindows applications. Support files are supplied for the following VAX programming languages: Ada, BASIC, BLISS, C, FORTRAN, MACRO, Pascal, and PL/1. In addition, support files are supplied for programming in the C language using conventions and standards developed at the Massachusetts Institute of Technology.
- Example files—The VMS DECwindows example files provide examples of files used to develop various DECwindows applications.

11 The procedure gives you the opportunity to change your mind before proceeding by displaying the following message:

```
* Is this correct? (Y/N)
```

If you answer YES to this prompt, the installation procedure begins restoring the VMS *required* save set. If you answer NO to this prompt, you can respecify the VMS and DECwindows options that you want to install.

The VMS DECwindows files are listed in Appendix C. Use DECW$TAILOR to remove the VMS DECwindows files you do not need. For complete information on using DECW$TAILOR to remove VMS DECwindows files see Chapter 12.

The procedure will restore the VMS DECwindows options you choose during Phase 2.

12 The procedure restores the VMS *required* save set and purges the page, swap, dump, and authorization files.

13 After several minutes, the upgrade procedure indicates that the system will shut down to reboot the partially upgraded VMS Version 5.4 system and displays information about how to reboot the system.

On certain VAX computers, the procedure displays the following message:

```
* Press RETURN to shutdown system:
```

Once you have read the information displayed on the screen, press Return to begin the shutdown.

14 If your system uses console media, it should automatically reboot from the SYSF directory when the shutdown is complete. If you have a MicroVAX or if your system does not reboot from SYSF, halt the system and boot from the [SYSF] directory. For most MicroVAX and VAXstation computers, use the BOOT command in the following format and substitute the device name of the system disk for *ddcu*:

```
>>> B/F0000000 ddcu
```

For more information on booting your computer from the SYSF directory, see the VMS upgrade and installation supplement for your VAX computer.

When the system reboots, it displays the VMS banner similar to the following:

```
VAX/VMS Version BC54-xxx Major version id = 1 Minor version id = 0
```

After the system reboots, the procedure continues with upgrade Phase 2.

8.5 Upgrade Phase 2

This section describes Phase 2 of the upgrade. During Phase 2, the procedure installs the VMS *library* and *optional* save sets (if you elected to install them), and VMS DECwindows software (if you elected to install it). You will be required to provide input during Phase 2 if your VMS kit is on more than one volume, or if you chose to install DECwindows options.

At the beginning of Phase 2, the procedure displays a message similar to the following:

```
Continuing with VAX/VMS V5.4 Upgrade Procedure.

         Upgrade Phase 2         19-JUN-1990 15:55:42.57
```

During Phase 2, the upgrade procedure does the following:

1. Phase 2 restores the rest of the optional VMS files. If your VMS distribution kit contains more than one piece of media, the procedure asks you to replace the volume in the drive with the next one in the kit.
2. After restoring the VMS files, the procedure displays a message similiar to the following:

   ```
   You may now remove the VMS distribution kit from DMA0:.
   ```

   Remove the VMS distribution media from the drive.
3. If you are not installing VMS DECwindows software, go to Section 8.6.

   If you are installing VMS DECwindows software, the procedure displays the following message:

```
Load the DECwindows distribution kit.

* Enter name of drive holding the DECwindows distribution media:
```

Note: If you have an RL02 kit, VMS DECwindows is on magnetic tape. If you have an RX33 kit, VMS DECwindows is on a tape cartridge. Check the labels and place the appropriate piece of media on the appropriate drive.

Insert the VMS DECwindows distribution media. Then enter the device name of the drive. The procedure displays the following message:

```
* Is the DECwindows media ready to be mounted? [N]
```

Type Y and press the Return key.

4. After restoring the VMS DECwindows files, the procedure displays a message similiar to the following:

   ```
   You may now remove the DECwindows distribution kit from MUA0:
   ```
5. Remove the VMS DECwindows distribution media from the drive.

8.6 Upgrade Phase 3

At the beginning of Phase 3, the procedure displays a message similar to the following:

```
Continuing with VAX/VMS V5.4, Upgrade Procedure

     Upgrade Phase 3 19-JUN-1990 16:45:18.35
```

During Phase 3, the upgrade procedure does the following:

- Merges the VMS-distributed files that are commonly edited by system managers with new VMS files.
- Merges all the miscellaneous user files in the old system directories into a new set of system directories, temporarily called SYSF.SYSEXE, SYSF.SYSMGR, SYSF.SYSLIB, and so on.
- Removes the directory entries for page, swap, dump, and authorization files from the old directory tree.

- Deletes all the remaining accounting data files, operator logs, and all files in the SYSERR directory.
- Merges the old and new versions of the following files: HELP, DCLTABLES, IMAGELIB.OLB, and STARLET.OLB.
- Moves all remaining files from the old directory tree [VMS$COMMON] to the new directory tree.
- Deletes the old directory tree [VMS$COMMON].

During Phase 3, the system may display LIBRAR-E-DUPGLOBAL errors and file-not-found messages which you can ignore.

8.7 Upgrade Phase 4

At the beginning of Phase 4, the procedure displays a message similar to the following:

```
Continuing with VAX/VMS V5.4 Upgrade Procedure.

     Upgrade Phase 4  19-JUN-1990 17:14:34.12
```

During Phase 4, the upgrade procedure does the following:

1. Corrects back pointers for system directories.
2. Shuts down the system.
3. May require you to manually reboot the system from the original system root (usually SYS0).

   If you are upgrading a VAX 8530, 8550, 8810, or 8820-N computer, do the following:

   a. Restore the original copies of DEFBOO and *ddd*GEN to boot from the original root directory.

   b. In the pre-upgrade procedure (Chapter 6), you made copies of the boot command procedures, DEFBOO.SAV and *ddd*GEN.SAV. To restore the original names to these files, enter the following commands and press the Return key after each one:

   ```
   >>>  EXIT
   $  COPY DEFBOO.SAV DEFBOO.COM
   $  COPY dddGEN.SAV dddGEN.COM
   ```

   For more information on booting the system, see the VMS upgrade and installation supplement for your VAX computer.

   For most MicroVAX and VAXstation computers, use the BOOT command in the following format and substitute the device name of the system disk for *ddcu*:

   ```
   >>>  B ddcu
   ```

4 When the system boots, the procedure displays a message similar to the following:

```
VAX/VMS Version BC54-xxx Major version id = 1 Minor version id = 0
```

After the system reboots, the procedure continues with upgrade Phase 5. Do not move the system disk to a different drive. The system disk must remain in the same drive for the procedure to continue with Phase 5.

8.8 Upgrade Phase 5

At the beginning of Phase 5, the procedure displays a message similar to the following:

```
Continuing with VAX/VMS V5.4 Upgrade Procedure.

     Upgrade Phase 5       19-JUN-1990 17:45:24.03
```

During Phase 5, the upgrade procedure does the following:

- Deletes the temporary [SYSF] directory tree.
- Cleans up files used only during the upgrade procedure.
- Converts all VAXVMSSYS.PAR parameter files found in system roots to Version 5.4 format.
- Checks the security of DECnet–VAX objects and images.
- Displays messages listing tasks that you might want to perform after the upgrade procedure is completed. Chapter 9 explains these tasks.

8.9 Upgrade Phase 6

During phase 6, the upgrade procedure applies the mandatory update, which is provided on a separate disk or tape. Use the following procedure to apply the mandatory update:

1 The procedure asks for the name of the drive that holds the disk or tape:

```
* Where will the distribution volumes be mounted:
```

Enter the device name of the drive. For example, if the mandatory update is on drive DJA1, enter the following:

```
* Where will the distribution volumes be mounted:  DJA1
```

The procedure asks you to mount the disk or tape. Make sure the disk or tape is write-protected and place it in the drive.

```
* Please mount the first volume of the set on ddcu:.
```

2 The procedure asks if you are ready. Type Y and press the Return key. For example:

```
* Are you ready?  Y
```

3 The procedure displays the following series of messages:

```
%MOUNT-I-MOUNTED, VMSMUP mounted on _DJA1:

The following products will be processed:

   VMSMUP V5.4

        Beginning installation of VMSMUP V5.4 at 15:00

%VMSINSTAL-I-RESTORE, Restoring product saveset A...
   .
   .
   .
```

4 If there is more than one disk or tape containing the mandatory update, you will be asked to switch them at some point during the procedure. Read the screen displays. When the procedure finishes, it displays a message similar to the following:

```
Installation of VMSMUP V5.4 completed at 15:00

VMSINSTAL procedure done at 15:02
```

After installing the mandatory update, the procedure restores your site-specific versions of the following system files:

```
[SYSMGR]DECW$SYLOGIN.COM
[SYSMGR]LOGIN.COM
[SYSMGR]LTLOAD.COM
[SYSMGR]RTTLOAD.COM
[SYSMGR]STARTNET.COM
[SYSMGR]SYLOGIN.COM
[SYSMGR]SYSTARTUP_V5.COM
[SYSMGR]SYCONFIG.COM
[SYSMGR]SYSECURITY.COM
[SYSMGR]SYSHUTDWN.COM
[SYSMGR]SYPAGSWPFILES.COM
[SYSMGR]SYLOGICALS.COM
[SYSMGR]TFF$STARTUP.COM
```

You might not have site-specific versions of all of these files. The procedure will display a message listing each file as it is restored.

During Phase 6, the upgrade procedure also gives you the opportunity to register any additional licenses and runs AUTOGEN.

1 The procedure displays the following message:

```
    If you have Product Authorization Keys (PAKs) to register, you
    may register them now.

* Do you want to register any Product Authorization Keys? (Y/N):
```

You do not have to re-register your VMS license or licenses for the following system integrated products:

VAXclusters
DECnet–VAX
RMS Journaling
Volume Shadowing

If you want to register a *new* Product Authorization Key, type Y and press the Return key. Otherwise, type N and press Return.

For more information on registering licenses, see the description of VMSLICENSE.COM in the *VMS Version 5.4 Release Notes*.

2 Next, the procedure displays the following message and runs AUTOGEN:

```
Running AUTOGEN to compute the new SYSGEN parameters...
```

If AUTOGEN encounters any problems, it displays the following message:

```
%AUTOGEN-W-REPORT, Warnings were detected by AUTOGEN. Please review the
        information given in the file SYS$SYSTEM:AGEN$PARAMS.REPORT.
```

If AUTOGEN displays this message, be sure to check AGEN$PARAMS.REPORT as one of your post-installation tasks. This task is described in Section 9.3.

3 After AUTOGEN finishes, the procedure displays a sequence of shutdown messages that begins like this:

```
The system is shutting down to allow the system to boot with the generated
site-specific parameters and installed images.

The system will automatically reboot after the shutdown and the
upgrade will be complete.

          SHUTDOWN -- Perform an Orderly System Shutdown
    .
    .
    .
```

4 The system shuts down and may automatically reboot (if your system is set up to do so). If the system does not automatically reboot, reboot it manually. For more information on booting your system, see the VMS upgrade and installation supplement for your VAX computer.

5 If you are upgrading a standalone system, skip this step.

If you are upgrading a VAXcluster system, the system might hang when you try to reboot the upgraded system disk. This is because the total cluster votes is less than the cluster quorum value. To remedy this situation, boot voting nodes that run from the system disk you just upgraded.

6 After the system reboots, the procedure displays the following message:

```
************************************************************

VAX/VMS V5.4

You have SUCCESSFULLY installed the VMS operating system.

The system is now executing the STARTUP procedure. Please
wait for the completion of STARTUP before logging in to the
system.

************************************************************
```

Finally, the procedure displays informational messages as well as accounting information. For example:

```
Startup processing continuing...

%SET-I-INTSET, login interactive limit=64, current interactive value = 0
 19-JUN-1990 18:30:00.00
 SYSTEM       job terminated at 19-JUN-1990 18:30:00.00

Accounting information:
 Buffered I/O count:        133   Peak working set size:        401
 Direct I/O count:           12   Peak virtual size:           2379
 Page faults:               325   Mounted volumes:                0
 Charged CPU time 0 00:00:55.23   Elapsed time        0 00:01:31.24
```

At this point the VMS operating system is running.

7 Press the Return key. The system asks for the user name and password. Log in to the SYSTEM account so that you can perform the post-upgrade procedures described in Chapter 9.

```
     Welcome to VAX/VMS V5.4
USERNAME:  SYSTEM
PASSWORD:
   .
   .
   .
Welcome to VAX/VMS V5.4
```

If you forget the password, follow the instructions for performing an emergency startup in the *Guide to Setting Up a VMS System*.

8 Go to Chapter 9 to perform the required post-upgrade tasks.

9 After You Do an Upgrade

After you upgrade the VMS operating system, you need to perform several important tasks before you can use the system. This chapter tells you what the tasks are and the order in which you perform them.

9.1 Post-Upgrade Checklist

Experienced system managers can use the following checklist to make sure they have performed all the necessary tasks. System Managers with less experience are advised to go to Section 9.2 for complete step-by-step instructions, and to use this checklist after they have completed the instructions in this chapter.

- ☐ Replace SYSUAF.DAT (described in Section 9.2).
- ☐ Examine AUTOGEN output stored in the file AGEN$PARAMS.REPORT (described in Section 9.3).
- ☐ Examine the command procedure templates supplied with VMS Version 5.4 (described in Section 9.5).
- ☐ Create a transaction log file, or disable the DECdtm software (described in Section 9.6), depending on whether or not you want to use DECdtm services.
- ☐ Examine MODPARAMS.DAT (described in Section 9.7).
- ☐ Run DECnet–VAX (described in Section 9.8).
- ☐ Update your console media (described in Section 9.9).
- ☐ Decompress the system libraries using LIBDECOMP.COM (described in Section 9.10).
- ☐ Set up the system messages help library and decompress the library, if desired (described in Section 9.11).
- ☐ Run the User Environment Test Package (UETP) to test the system (described in Chapter 10).
- ☐ Remove unwanted files and check the sizes of system files (described in Section 9.13).
- ☐ Customize VMS DECwindows software (described in Chapter 11).
- ☐ Build a standalone BACKUP kit (described in Section 9.15).
- ☐ Back up the console media (described in Section 9.15).
- ☐ Back up the system disk (described in the VMS upgrade and installation supplement for your computer).
- ☐ Run AUTOGEN (described in Section 9.16).

9.2 Replacing SYSUAF.DAT

If you permanently store SYSUAF.DAT on the system disk, skip this section and go to Section 9.3.

If you do not permanently store SYSUAF.DAT on the system disk, you copied it to the system disk before you did the upgrade. The upgrade procedure updates SYSUAF.DAT. Copy this updated version of SYSUAF.DAT from the system disk to the directory where you usually keep it.

Note: If you did a rolling upgrade, the shared version of SYSUAF.DAT was not updated.

9.3 Examining AGEN$PARAMS.REPORT

When AUTOGEN runs, it writes informational and, if necessary, warning messages to the file SYS$SYSTEM:AGEN$PARAMS.REPORT. You should examine the contents of this report file.

To view AGEN$PARAMS.REPORT on your screen, type the following command and press the Return key:

```
$  TYPE SYS$SYSTEM:AGEN$PARAMS.REPORT
```

For more information on AGEN$PARAMS.REPORT, see the *VMS Version 5.4 New Features Manual*.

If the report includes a message similar to the following, you might need to modify the size of the PAGE, SWAP, or DUMP file:

```
******************
%AUTOGEN-W-DSKSPC, The disk on which DISK$VAXVMSRL5:[SYS0.SYSEXE]PAGEFILE.SYS
        resides would be over 95% full if it were modified to hold 20000
        blocks.
******************
```

For more information on modifying the sizes of the PAGE, SWAP, and DUMP files, see Section 9.4.

9.4 Checking the Size of System Files

AUTOGEN sets the following files at sizes appropriate for your system:

[SYSEXE]SYSDUMP.DMP
[SYSEXE]PAGEFILE.SYS
[SYSEXE]SWAPFILE.SYS

For special workloads or configurations, you may want to specify different sizes for these files. Check the sizes of these files to be sure they meet your needs. To modify the size of these files, log into the SYSTEM account and enter the following command:

```
$  @SYS$UPDATE:SWAPFILES
```

For more information, see the *Guide to Setting Up a VMS System*.

9.5 Examining Your Command Procedures

The upgrade procedure places a new version of WELCOME.TXT on your system disk. If your system uses a site-specific version of this file, you must modify the new version of the file with your changes.

The upgrade procedure restores your site-specific version of the following files:

```
[SYSMGR]DECW$SYLOGIN.COM
[SYSMGR]LOGIN.COM
[SYSMGR]LTLOAD.COM
[SYSMGR]RTTLOAD.COM
[SYSMGR]STARTNET.COM
[SYSMGR]SYLOGIN.COM
[SYSMGR]SYSTARTUP_V5.COM
[SYSMGR]SYCONFIG.COM
[SYSMGR]SYSECURITY.COM
[SYSMGR]SYSHUTDWN.COM
[SYSMGR]SYPAGSWPFILES.COM
[SYSMGR]SYLOGICALS.COM
[SYSMGR]TFF$STARTUP.COM
```

The upgrade procedure provides new templates for these files with the .TEMPLATE extension. The new templates might include features that are not in your site-specific files. Check the templates against your site-specific files and edit your files as necessary.

9.6 Customizing or Disabling DECdtm Services

By default, processes for DECdtm services are started when a full VMS boot is executed. The DECdtm process TP_SERVER then checks for the existence of a transaction log file on the system and continues checking every 15 seconds.

If you plan to run DECdtm software you must create a transaction log file before transactions can be started. To create a transaction log file, you use the LMCP utility. Refer to the LMCP section in the *VMS Version 5.4 New Features Manual* for information about creating a log file.

If you do not want to run DECdtm software, you can prevent the startup of DECdtm processes by defining the system-wide logical name SYS$DECDTM_INHIBIT in the SYS$MANAGER:SYLOGICALS.COM command procedure. You can define SYS$DECDTM_INHIBIT to be any string. For example:

```
$ DEFINE/SYSTEM/EXEC SYS$DECDTM_INHIBIT "yes"
```

For information on the SYLOGICALS.COM command procedure, see the *Guide to Setting Up a VMS System*.

9.7 Changing MODPARAMS.DAT

Review the file SYS$SYSTEM:MODPARAMS.DAT. The upgrade procedure created a new version of this file. The old version is named SYS$SYSTEM:MODPARAMS.DAT_OLD. Modify the new file, if necessary, for your system.

If you are upgrading a VAXcluster, you must update the MODPARAMS.DAT file for each VAX computer that boots from the system disk. Make sure the EXPECTED_VOTES value is correct. The value is the sum of all VOTES in the cluster. For example, if there are five VAX computers in the cluster, and each has one VOTE, the value is five. As you reboot each VAX computer, AUTOGEN runs automatically. The cluster forms when you have booted enough computers to attain cluster quorum.

9.8 Updating Your DECnet–VAX Configuration

If you are not running DECnet–VAX, skip this section and go to Section 9.9.

If you are running DECnet–VAX, follow the instructions in Section 9.8.1 and Section 9.8.2.

9.8.1 Running NETCONFIG_UPDATE.COM

As of VMS Version 5.2, DECnet–VAX provides a more secure network environment by default. During Phase 5, the upgrade procedure checks the security of DECnet–VAX objects and images on your system. If your DECnet–VAX configuration does not meet the default level of security, the procedure displays the following message recommending that you run the NETCONFIG_UPDATE command procedure.

```
The upgrade has detected items which should be reviewed to ensure that
DECnet-VAX will operate properly and securely.

Once the upgrade completes, you should manually review the items listed
above.  Failure to do so could leave your node vulnerable to certain
forms of attack.  For futher information, consult the "VMS Guide to
System Security," Chapter 7.

You can execute the command procedure SYS$UPDATE:NETCONFIG_UPDATE.COM
to update the security of your existing DECnet configuration. For more
information on NETCONFIG_UPDATE.COM see the VMS Upgrade and Installation
Manual.
```

NETCONFIG_UPDATE.COM is a command procedure introduced with VMS Version 5.2 that implements several security measures that affect default access to your system. When you run NETCONFIG_UPDATE.COM, your configuration database does not change, except for the changes you may make to the default DECNET account or the accounts you may create for specific objects.

You can only use NETCONFIG_UPDATE.COM to *increase* your level of network security. The procedure will not prompt you for input if your network configuration meets the minimum level established with VMS Version 5.2.

NETCONFIG_UPDATE.COM generates passwords for each account that you decide to keep in your configuration database. For example, if you currently have a default DECNET account and decide to keep it, NETCONFIG_UPDATE.COM generates a new password for it and replaces the existing password with the new one.

The following steps show an example of running NETCONFIG_ UPDATE.COM:

Note: Depending on your network configuration, the procedure might not display all of the questions shown in this example.

1 Enter the following command:

```
$ @SYS$UPDATE:NETCONFIG_UPDATE
```

The procedure displays the following message:

```
     DECnet--VAX node configuration upgrade procedure

     This procedure will allow you to create, display and
     optionally execute the necessary commands to update your
     executor and object database characteristics so that
     they will be compatible with V5.2 of DECnet--VAX.


     Getting Executor Information
     Getting Object Information

     Modifying Executor node 6.36 (KRUP)
Do you wish a default nonprivileged DECnet account?  [NO]:
```

2 The default DECnet account, named DECNET, allows all network objects general access to the system. It is only appropriate for networks with very low security requirements. For networks with greater security requirements, Digital recommends separate default accounts for specific network objects. After you respond to this prompt, the procedure displays the following message:

```
Do you wish to disable the TASK object?  [YES]:
```

3 The TASK object, by means of the default DECnet account (DECNET), allows user-written command procedures—including those which might be used in attempted break-ins—to be executed on your system. You can disable default access to the TASK object and still allow remote user-written command procedures to run on your system by using access control strings or proxy access. After you respond to this prompt, the procedure displays the following message:

```
Do you want a default account for the MAIL object?  [YES]:
```

4 MAIL is an image that provides personal mail services for VMS systems. To receive MAIL, you must create a default account for the MAIL object. An account named MAIL$SERVER will be created on your system if you reply YES to this query. After you respond to this prompt, the procedure displays the following message:

```
Do you want a default account for the FAL object?  [NO]:
```

5 FAL (File Access Listener) is the remote file access facility. FAL is an image that receives and processes remote file access requests for files at the local node. A default account for FAL allows network access, by any remote user, to any files marked world-accessible. It also allows any remote user to create files in any directory with world WRITE access. An account named FAL$SERVER will be created on your system if you reply YES to this query. After you respond to this prompt, the procedure displays the following message:

```
Do you want a default account for the PHONE object?  [YES]:
```

6 PHONE is an image that allows online conversations with users on remote VMS systems. A default account for this object lets anyone in the network get a list of users currently logged in to the local system. A user can then attempt a login using the list of user names. An account named PHONE$SERVER will be created on your system if you reply YES to this query. After you respond to this prompt, the procedure displays the following message:

```
Do you want a default account for the NML object?  [YES]:
```

7 NML is the Network Management Listener. A default account for this object lets remote users issue NCP TELL commands to gather and report network information from your volatile DECnet–VAX databases. An account named NML$SERVER will be created on your system if you reply YES to this query. After you respond to this prompt, the procedure displays the following message:

```
Do you want a default account for the VPM object?  [YES]:
```

8 VPM is the VMS Performance Monitor. To use the cluster monitoring features of the Monitor Utility (MONITOR), you must create a default VPM account. An account named VPM$SERVER will be created on your system if you reply YES to this query. After you respond to this prompt, the procedure displays the following message:

```
Do you want a default account for the MIRROR object?  [YES]:
```

9 MIRROR is an image that is used for most types of loopback tests, including those run during the DECnet phase of the User Environment Test Program (UETP). To run most loopback tests, you must create a default account for the MIRROR object. An account named MIRRO$SERVER will be created on your system if you reply YES to this query.

10 Next the procedure displays a series of commands and the following message:

```
Do you wish to execute these commands now?  [YES]:
```

11 If you execute the commands, the procedure displays the following message:

```
        The DECnet--VAX database has been updated
```

For a description of the network configuration procedure, NETCONFIG.COM, refer to the *Guide to DECnet–VAX Networking* or to the *VMS Networking Manual*. For information on the security enhancements to NETCONFIG.COM, refer to the *VMS Version 5.2 New Features Manual*.

9.8.2 Running UPDATE_CLUSTER_MEMBERS.COM

When you run NETCONFIG_UPDATE.COM on a member of a VAXcluster environment, the NETCONFIG_UPDATE.COM procedure creates the secondary procedure named SYS$COMMON:[SYSMGR]UPDATE_CLUSTER_MEMBERS.COM. NETCONFIG_UPDATE.COM instructs you to run UPDATE_CLUSTER_MEMBERS.COM on the other VAXcluster members. This secondary procedure modifies the default access of each VAXcluster member exactly as you modified that of the first member.

To run UPDATE_CLUSTER_MEMBERS.COM using SYSMAN, do the following:

```
$  RUN SYS$SYSTEM:SYSMAN
SYSMAN>  SET ENVIRONMENT/CLUSTER/USER=SYSTEM
     Remote Password:

     %SYSMAN-I-ENV, current command environment:
             Clusterwide on local cluster
             Username SYSTEM will be used on nonlocal nodes
SYSMAN>  DO @SYS$COMMON:[SYSMGR]UPDATE_CLUSTER_MEMBERS.COM
SYSMAN>  EXIT
```

9.9 Updating Your Console Media

If your computer does not use console media, skip this section and go to Section 9.10.

If your computer uses console media, you must copy the Version 5.4 VMB.EXE onto your system's console media. Section 9.9.1 describes the procedure to use if you have a VAX 8530, 8550, 8810 (8700), or 8820-N (8800) computer.

Section 9.9.2 describes the procedure to use if you have one of the following VAX computers:

VAX–11/730
VAX–11/750
VAX–11/780, 785
VAX 8200, 8250, 8300, 8350
VAX 8600, 8650

Section 9.9.3 contains special information for the VAX 9000 computer.

9.9.1 VAX 8530, 8550, 8810 (8700), 8820-N (8800)

If you have a a VAX 8530, 8550, 8810 (8700), or 8820-N (8800), follow the directions in this section.

1. Place a blank RX50 diskette in one of the console diskette drives. The console refers to the upper (or left-hand) diskette drive as DZ1 while the VMS operating system refers to it as CSA1. The console refers to the lower (or right-hand) drive as DZ2 while the VMS operating system refers to it as CSA2.

2. Make sure the diskette drives have been connected to the VMS operating system, as follows:

```
$  SHOW DEVICE CSA
Device                  Device           Error     Volume
 Name                   Status           Count      Label
CSA0:          (SYSTEM)  Online               0
CSA1:          (SYSTEM)  Online               0
CSA2:          (SYSTEM)  Online               0
```

   If the drives are not connected, the following message is displayed:

```
%SYSTEM-W-NOSUCHDEV, no such device available
```

   If this message is displayed, enter the following commands (you need CMKRNL privilege):

```
$  RUN SYS$SYSTEM:SYSGEN
SYSGEN>  CONNECT CONSOLE
SYSGEN>  EXIT
```

3. To initialize the diskette, enter a command in the following format and press the Return key:

```
$  INITIALIZE/STRUCTURE=1 device-name: volume-label
```

   Substitute CSA1 or CSA2 for *device-name*. Substitute a 1- to 12-character name for *volume-label*. For example:

```
$  INITIALIZE/STRUCTURE=1 CSA1: V54CONSOLE
```

4. Enter the following commands to mount the diskette, create the required directories, and copy the new VMB.EXE to the diskette. Substitute CSA1 or CSA2 for *device-name*.

```
$  MOUNT/OVERRIDE=ID device-name:
$  CREATE/DIRECTORY device-name:[TRANSFER]
$  COPY SYS$SYSTEM:VMB.EXE device-name:[TRANSFER]
```

5. Once you have copied all the files, enter the following command to dismount the diskette. Substitute CSA1 or CSA2 for *device-name*.

```
$  DISMOUNT device-name:
```

6. To enter console mode, press CTRL/P. To exit from the console program, enter the following command and press Return:

```
>>>  EXIT
```

7. Open the diskette drive door, pause for a moment, and close it. The red indicator light should flash.

8 To copy VMB.EXE from the diskette to the console fixed disk drive, enter the following command and press Return:

```
$  COPY DZu:[TRANSFER]VMB.EXE   LB0:[CONSOLE]
```

Substitute DZ1 or DZ2 for DZu.

The fixed drive in the console is known by two names, DW2 and LB0. Most console files are stored using the name LB0. If the system displays a message that describes a protection violation on the output device, copy the files using the device name DW2.

9 When you have finished copying the files, remove the diskette from the drive.

10 Restart the console program by entering the following command and pressing Return:

```
$  RUN CONTROL
```

11 Return to the VMS operating system by entering the following command and pressing Return:

```
>>>  SET TERM PROGRAM
```

9.9.2 VAX 11/730, 750, 780, 785 and VAX 8200, 8250, 8300, 8350, 8600, 8650

If you have one of the following VAX computers, follow the directions in this section:

VAX–11/730
VAX–11/750
VAX–11/780, 785
VAX 8200, 8250, 8300, 8350
VAX 8600, 8650

This procedure does the following:

- Checks to see if you have enough space for the new VMB.EXE
- Copies the new VMB.EXE to the console media
- Updates the BOOT58 file for the VAX 11/750, 8200, 8250, 8300 and 8350 computers

To update the console media, use the following procedure:

1 Log into the SYSTEM account.

2 Enter the following commands to check if there is enough space for the new VMB.EXE. You need 120 free blocks. If you are updating a VAX–11/730, use CSA2 for *device-name*; if you are updating one of the other VAX computers, use CSA1 for *device-name*.

```
$  RUN SYS$SYSTEM:SYSGEN
SYSGEN>  CONNECT CONSOLE
SYSGEN>  EXIT
$  EXCHANGE DIRECTORY device-name:VMB.EXE

Directory of RT-11 volume TINES$CSA1:      19-APR-1990 14:04

  VMB.EXE      80   19-Apr-1990
```

```
Total of 1 file, 80 blocks.  Free space 43 blocks, largest 23.
```

If the size of the current VMB, plus the free space, does not equal at least 120 blocks, delete unneeded files on the console until you have 120 blocks available. In the above example, the size of the current VMB.EXE (80 blocks) plus the free space (43 blocks) equals 123 blocks, so files do not have to be deleted.

3 Enter the following command and press the Return key:

```
$ @SYS$UPDATE:UPDATE_CONSOLE.COM
```

4 If you are updating a VAX 8600 or VAX 8650 computer, the new VMB.EXE is copied onto the console.

If you are updating one of the other VAX computers, the procedure uses the EXCHANGE procedure to copy the contents of the existing console, merge the new files onto the copy of the console, and create a new version of the console media. When the computer is ready to create the new version, it displays a message asking you to insert a scratch tape cassette or diskette. Take out the original console media and insert a new one.

9.9.3 VAX 9000

VAX 9000 computers use a special version of VMB.EXE named VMB9AQ.EXE. The upgrade procedure automatically copies a new version of this file to the console hard disk. However, if for some reason you need to update the console hard disk, use the UPDATE_CONSOLE.COM command procedure described in Section 9.9.2.

9.10 Decompressing the System Libraries

Decompressing the system libraries gives the system faster access to the libraries. The decompressed libraries require approximately 8500 additional blocks of disk space. To find out how much disk space you have, enter the following command and press the Return key:

```
$ SHOW DEVICE SYS$SYSDEVICE
```

If you have enough room on the disk, you can decompress the libraries. The decompression process takes approximately 30 minutes. Log into the SYSTEM account, enter the following command, and press Return:

```
$ @SYS$UPDATE:LIBDECOMP
```

9.11 Setting Up the System Messages Help Library

For VMS Version 5.4, you can install and access an optional online Help version of the *VMS System Messages and Recovery Procedures Reference Manual*. Because this is a large file, it is not included as part of the default root library, SYS$HELP:HELPLIB.HLB. Instead, it is provided as a separate file named SYS$HELP:SYSMSGHELP.HLB. You can view the contents of this file by accessing it in one of three ways:

- Use the /LIBRARY qualifier with the HELP command. For example:

```
$  HELP/LIBRARY=SYS$HELP:SYSMSGHELP.HLB ERRORS ACCVIO
```

- Or, define a logical name that instructs the Help facility to search the new help library when it does not find the specified topic in the VMS root help library. For example:

  ```
  $  DEFINE HLP$LIBRARY DISK$2:[QUAIL]SYSMSGHELP
  $  HELP ERRORS DISMAL
  ```

  In this example, the DEFINE statement creates a logical name for the help library that the Help facility is to search after it has searched the root library, SYS$HELP:HELPLIB.HLB. In this example, the help library is located in DISK$2:[QUAIL]; you should specify the appropriate disk and directory for your system.

  First, the Help facility searches the root library for ERRORS. If it does not find ERRORS,[1] it searches the library defined by HLP$LIBRARY until it finds ERRORS and displays the appropriate information. For information on defining logical names and search patterns for the Help facility, see the HELP command in the *VMS DCL Dictionary*.

- Or, using the VMS Librarian Utility, you can extract the ERRORS module from SYSMSGHELP.HLB and insert it into the default root help library HELPLIB.HLB. This allows direct access without using extra HELP qualifiers or logical names. For more information, see the *VMS Librarian Utility Manual*.

The system messages help library is in compressed format. Decompressing the library gives you faster access to it but requires an additional 1600 blocks of disk space. To decompress the library, enter a command similiar to the following:

```
$  LIBRARY/DATA=EXPAND/OUTPUT=device:[directory]SYSMSGHELP.HLB -
_$  device:[directory]SYSMSGHELP.HLB
```

In this example, *device* is the name of the device where the file is located, and *directory* is the name of the directory.

Note: The system messages help library is not decompressed when you execute the LIBDECOMP.COM procedure described in Section 9.10.

The system messages help library can be deleted without affecting other help libraries. You can also add or delete this library using the VMS tailoring utility (VMSTAILOR).

9.12 Testing the System with UETP

Run the User Environment Test Package (UETP) to verify that the upgrade was done correctly. For instructions on running UETP, see Chapter 10.

[1] Previous versions of HELPLIB.HLB provided information about system messages format under the name ERROR. This information is now named FORMAT_OF_ERROR.

9.13 Increasing Free Space on the System Disk

This section describes how to increase the amount of free space on your system disk. You can gain free space by removing optional VMS or VMS DECwindows files that you do not need from the system disk or by decreasing the size of some system files.

9.13.1 Using VMSTAILOR and DECW$TAILOR

To remove VMS files that you do not need from the system disk, use VMSTAILOR. For example, if you are not running DECnet–VAX, you do not need the network support files. For complete information on using VMSTAILOR to remove VMS software see Chapter 12.

To remove unneeded DECwindows files from the system disk, use DECW$TAILOR. For example, if you do not plan to use the DECwindows programming support, you can remove those files. For complete information on using DECW$TAILOR to remove DECwindows files see Chapter 12.

9.13.2 Purging System Files

You might want to purge the following VMS files to increase disk space:

```
SYS$SYSTEM:SHUTDOWN.COM
SYS$SYSTEM:STARTUP.COM
SYS$LIBRARY:ENCRYPSHR.EXE
SYS$LIBRARY:*RTL*.EXE
SYS$LIBRARY:*SHR.EXE
SYS$MANAGER:EDTINI.EDT
SYS$MANAGER:DBLSTRTUP.COM
SYS$MANAGER:*.TEMPLATE
SYS$EXAMPLES:XADRIVER.MAR
SYS$LIBRARY:CDDSHR.EXE
SYS$LIBRARY:EPC$SHR.EXE
SYS$LIBRARY:EPC$FACILITY.TLB
SYS$LIBRARY:EPC$MSG.EXE
SYS$LIBRARY:UISSHR.EXE
SYS$STARTUP:VMS$LAYERED.DAT
```

To purge a file, enter the PURGE command followed by the file specification. For example:

```
$  PURGE SYS$SYSTEM:SHUTDOWN.COM
```

9.14 Customizing VMS DECwindows Software

If you have installed VMS DECwindows software, there are several steps that you must take to customize the VMS DECwindows environment. Chapter 11 describes these steps. After you finish customizing VMS DECwindows software, return to this chapter.

9.15 Backing Up the System Disk

Digital recommends that you protect your work by performing the following operations:

- Make a standalone BACKUP kit
- Back up the console media (if applicable)
- Back up the system disk

For complete information on these operations, see the VMS upgrade and installation supplement for your VAX computer.

9.16 Running AUTOGEN

After 24 hours of operation, run AUTOGEN in FEEDBACK mode and reboot the system. Run AUTOGEN in this way again 2 workdays later. AUTOGEN sets the values of system parameters and the sizes of the page and swap files according to the system's workload. Digital recommends that you run AUTOGEN from SAVPARAMS through TESTFILES on a weekly basis and examine AGEN$PARAMS.REPORT to determine the need for additional changes.

Hardcoded values in MODPARAMS.DAT affect AUTOGEN's calculations of the feedback parameters. For AUTOGEN FEEDBACK to properly calculate values, you should replace the hardcoded values in MODPARAMS.DAT with MIN_ values.

For information on using AUTOGEN, see the *Guide to Setting Up a VMS System*.

Part IV

Part IV describes additional tasks you might perform after installing or upgrading the VMS operating system.

10 Running UETP

The User Environment Test Package (UETP) is a VMS software package designed to test whether the VMS operating system is installed correctly. UETP puts the system through a series of tests that simulate a typical user environment, by making demands on the system that are similar to demands that might occur in everyday use.

UETP is not a diagnostic program; it does not attempt to test every feature exhaustively. When UETP runs to completion without encountering nonrecoverable errors, the system being tested is ready for use.

UETP exercises devices and functions that are common to all VMS systems, with the exception of optional features such as high-level language compilers. The system components tested include the following:

- Most standard peripheral devices
- The system's multiuser capability
- DECnet–VAX
- Clusterwide file access and locks

10.1 Summary of UETP Operating Instructions

This section summarizes the procedure for running all phases of UETP with default values. If you are familiar with the test package, refer to this section. If you need further information, refer to Section 10.2.

1 Log in to the SYSTEST account as follows:

```
Username:  SYSTEST
Password:
```

Note: Because the SYSTEST and SYSTEST_CLIG accounts have privileges, unauthorized use of these accounts might compromise the security of your system.

2 Make sure no user programs are running and no user volumes are mounted. By design, UETP assumes and requests the exclusive use of system resources. Unpredictable results could occur if you ignore this restriction.

3 After you log in, check all devices to be sure that the following conditions exist:

- All devices you want to test are powered up and are on line to the system.
- Scratch disks are mounted and initialized.
- Disks contain a directory named [SYSTEST] with OWNER_ UIC=[1,7]. (You can create this directory with the DCL command CREATE/DIRECTORY.)

- Scratch magnetic tape reels are *physically* mounted on each drive you want tested and are initialized with the label UETP (using the DCL command INITIALIZE). Make sure magnetic tape reels contain at least 600 feet of tape.
- Scratch tape cartridges have been inserted in each drive you want to test and are initialized with the label UETP.
- Line printers and hardcopy terminals have plenty of paper.
- Terminal characteristics and baud rate are set correctly (see the user's guide for your terminal).

Note that some communications devices need to be set up by Digital Customer Service (see Section 10.3).

If you encounter any problems in preparing to run UETP, read Section 10.3 before proceeding.

4 To start UETP, enter the following command and press RETURN:

```
$ @UETP
```

UETP responds with the following question:

```
Run "ALL" UETP phases or a "SUBSET" [ALL]?
```

Press RETURN to choose the default response enclosed in brackets. UETP responds with three more questions in the following sequence:

```
How many passes of UETP do you wish to run [1]?
How many simulated user loads do you want [n]?
Do you want Long or Short report format [Long]?
```

Press RETURN after each prompt. After you answer the last question, UETP initiates its entire sequence of tests, which run to completion without further input. The final message should look like the following:

```
**************************************************
*                                                *
   END OF UETP PASS 1 AT 19-JUN-1990 16:30:09.38
*                                                *
**************************************************
```

5 After UETP runs, check the log files for errors. If testing completes successfully, the VMS operating system is in proper working order.

Note: After a run of UETP, you should always run the Error Log Utility to check for hardware problems that can occur during a run of UETP. For information on running the Error Log Utility, refer to the *VMS Error Log Utility Manual*.

If UETP does not complete successfully, refer to Section 10.6.

If you want to run UETP without using the default responses, refer to Section 10.5 through Section 10.5.4, which explain the options.

10.2 Logging In

This section contains detailed instructions on running UETP. For a brief summary of these steps, see Section 10.1.

Obtain the SYSTEST password from your system manager. Log into the SYSTEST account from the console terminal as follows:

```
Username:  SYSTEST
Password:
```

Note: Because SYSTEST has privileges, unauthorized use of this account might compromise the security of your system.

UETP will fail if you do not run the test from the SYSTEST account. Also, if you try to run UETP from a terminal other than the console terminal, the device test phase displays an error message stating that the terminal you are using is unavailable for testing. You can ignore this message.

After you log into the SYSTEST account, enter the command SHOW USERS to make sure no user programs are running and no user volumes are mounted. UETP requires exclusive use of system resources. If you ignore this restriction, UETP may interfere with applications that depend on these resources.

10.2.1 SYSTEST Directories

If you logged in successfully, you should be in the root directory [SYSTEST] on the system disk. UETP uses directories named [SYSTEST] to hold all the files used by UETP command procedure (UETP.COM) and temporary files used by UETP during testing.

The DCL command SHOW LOGICAL displays the translation of the logical name SYS$TEST on a typical system:

```
$  SHOW LOGICAL SYS$TEST
   "SYS$TEST" = "SYS$SYSROOT:[SYSTEST]"  (LNM$SYSTEM_TABLE)
```

If you want UETP to test a particular disk, such as a scratch disk, create either a [SYSTEST] directory or a [SYS0.SYSTEST] directory on that disk. Section 10.3.2 discusses setting up scratch disks for testing.

10.3 Setting Up for UETP

After you log in, you need to set up the devices on the system for UETP testing.

Note: Your system may not have all the devices described in this section.

You should check all devices to be sure that the following conditions exist:

- All devices you want to test are turned on and are on line.
- Scratch disks are mounted and initialized.
- Disks contain a directory named [SYSTEST] with OWNER_UIC=[1,7]. Use the CREATE/DIRECTORY command if the [SYSTEST] directory does not exist on the disk.

- Scratch magnetic tape reels are *physically* mounted on each drive you want tested and are initialized with the label UETP (using the DCL command INITIALIZE). Make sure magnetic tape reels contain at least 600 feet of tape.
- Scratch tape cartridges have been inserted in each drive you want to test and are mounted and initialized with the label UETP.
- Line printers and hardcopy terminals have plenty of paper.
- Terminal characteristics and baud rate are set correctly (see the user's guide for your terminal).

Note that some communications devices discussed in this section need to be set up by Digital Customer Service.

10.3.1 The System Disk

Before running UETP, be sure that the system disk has at least 1200 blocks available. Note that systems running more than 20 load test processes may require a minimum of 2000 available blocks. If you run multiple passes of UETP, log files will accumulate in the default directory and further reduce the amount of disk space available for subsequent passes.

If disk quotas are enabled on the system disk, you should disable them before you run UETP.

10.3.2 Additional Disks

The disk test uses most of the available free space on each testable disk in the following manner:

- On each testable disk, the device test phase tries to create two files. The size of these files depends on how much free space is available on the disk. Usually the test creates each file with 5% of the free space on the disk. However, if the disk is nearly full, the test creates files that are 5 blocks. If the test cannot create 5 block files, it fails. Only the initial file creation can cause the device test to fail because of lack of disk space.
- The test randomly reads and writes blocks of data to the files. After every multiple of 20 writes for each file, the test tries to extend the file. The size of this extension is either 5% of the free disk space, or 5 blocks if the file was created with 5 blocks. This process of extension continues until the combined space of the files reaches 75% of the free disk space.

By creating and extending fragmented files in this way, UETP exercises the disk. This allows the test to check for exceeded quotas or a full disk, and to adjust for the amount of available disk space.

To prepare each disk drive in the system for UETP testing, use the following procedure:

1. Place a scratch disk in the drive and spin up the drive. If a scratch disk is not available, use any disk with a substantial amount of free space; UETP does not overwrite existing files on any volume. If your scratch disk contains files that you want to keep, do not initialize the disk; go to step 3.
2. If the disk does not contain files you want to save, initialize it. For example:

   ```
   $ INITIALIZE DUA1: TEST1
   ```

   This command initializes DUA1, and assigns the volume label TEST1 to the disk. All volumes must have unique labels.
3. Mount the disk. For example:

   ```
   $ MOUNT/SYSTEM DUA1: TEST1
   ```

   This command mounts the volume labeled TEST1 on DUA1. The /SYSTEM qualifier indicates that you are making the volume available to all users on the system.
4. UETP uses the [SYSTEST] directory when testing the disk. If the volume does not contain the directory [SYSTEST], you must create it. For example:

   ```
   $ CREATE/DIRECTORY/OWNER_UIC=[1,7] DUA1:[SYSTEST]
   ```

   This command creates a [SYSTEST] directory on DUA1 and assigns a user identification code (UIC) of [1,7]. The directory must have a UIC of [1,7] to run UETP.

If the disk you have mounted contains a root directory structure, you can create the [SYSTEST] directory in the [SYS0.] tree.

10.3.3 Magnetic Tape Drives

Set up magnetic tape drives that you want to test by doing the following:

1. Place a scratch magnetic tape with at least 600 feet of magnetic tape in the tape drive. Make sure that the write-enable ring is in place.
2. Position the magnetic tape at the beginning-of-tape (BOT) and put the drive on line.
3. Initialize each scratch magnetic tape with the label UETP. For example, if you have physically mounted a scratch magnetic tape on MTA1, enter the following command and press RETURN:

   ```
   $ INITIALIZE MTA1: UETP
   ```

   Magnetic tapes must be labeled UETP to be tested. As a safety feature, UETP does not test tapes that have been mounted with the MOUNT command.

If you encounter a problem initializing the magnetic tape, or if the test has a problem accessing the magnetic tape, refer to the description of the INITIALIZE command in the *VMS DCL Dictionary*.

10.3.4 Tape Cartridge Drives

Set up tape cartridge drives that you want to test by doing the following:

1. Insert a scratch tape cartridge in the tape cartridge drive.
2. Initialize the tape cartridge. For example:

```
$  INITIALIZE MUA0: UETP
```

   Tape cartridges must be labeled UETP to be tested. As a safety feature, UETP does not test tapes that have been mounted with the MOUNT command.

If you encounter a problem initializing the tape cartridge, or if the test has a problem accessing the tape cartridge, refer to the description of the DCL INITIALIZE command in the *VMS DCL Dictionary*.

10.3.4.1 TLZ04 Tape Drives

During the initialization phase, UETP sets a time limit of 6 minutes for a TLZ04 unit to complete the UETTAPE00 test. If the device does not complete the UETTAPE00 test within the alloted time, UETP displays a message similar to the following:

```
-UETP-E-TEXT, UETTAPE00.EXE testing controller MKA was stopped ($DELPRC) at 16:23:23.07
        because the time out period (UETP$INIT_TIMEOUT) expired or
        because it seemed hung or because UETINIT01 was aborted.
```

To increase the timeout value, type a command similiar to the following before running UETP:

```
$   DEFINE/GROUP UETP$INIT_TIMEOUT "0000 00:08:00.00"
```

This example defines the initialization timeout value to 8 minutes.

10.3.5 Compact Disc Drives

To run UETP on an RRD40 or RRD50 compact disc drive, you must first load the test disc that you received with your compact disc drive unit.

10.3.6 Optical Disk Drives

To run UETP on an RV60 drive, set up the RV64 optical disk-storage system, by doing the following:

1. Use the Jukebox Control Software (JCS) to load an optical disk in each of the RV60 drives. JCS is a layered product on the VMS operating system that comes with the RV64 and is responsible for controlling the robot arm that loads and unloads the disks.
2. Initialize the optical disks with the label **UETP**, but do not mount them.

UETP tests all the RV60s present in the RV64 simultaneously. Unlike the other tape tests, UETP does not re-initialize the optical disks at the end of the test.

10.3.7 Terminals and Line Printers

Terminals and line printers must be turned on to be tested by UETP. They must also be on line. Check that line printers and hardcopy terminals have enough paper. The amount of paper required depends on the number of UETP passes that you plan to execute. Each pass requires two pages for each line printer and hardcopy terminal.

Check that all terminals are set to the correct baud rate and are assigned appropriate characteristics (see the user's guide for your terminal).

Spooled devices and devices allocated to queues fail the initialization phase of UETP and are not tested.

10.3.8 Ethernet Adapters

Make sure that no other processes are sharing the Ethenet adaptor device when you run UETP. Because either DECnet–VAX or the LAT terminal server might also try to use the Ethernet adapter (a shareable device), you must shut down DECnet–VAX and the LAT terminal server before you run the device test phase, if you want to test the Ethernet adapter.

Note: UETP will not test your Ethernet adapter if DECnet–VAX or another application has the device allocated.

10.3.9 The DR11–W

Note: Only Digital Customer Service personnel should set up the DR11–W for UETP testing.

The DR11–W uses an internal logical loopback mode that tests all functionality except that of module connectors, cables, and transceivers. Because random external patterns are generated during this operation, the user device or other processor might need to be isolated from the DR11–W being tested until the testing is complete.

To test the DR11–W properly, the E105 switchpack must be set as follows:

Switch 1	Switch 2	Switch 3	Switch 4	Switch 5
Off	On	Off	Off	On

When UETP testing is completed, restore the DR11–W to the proper operating configuration.

10.3.10 The DRV11–WA

Note: Only Digital Customer Service personnel should set up the DRV11–WA for UETP testing.

To prepare the DRV11–WA on a MicroVAX for UETP testing, be sure the following conditions exist:

- The jumpers on the DRV11–WA board are set to W2, W3, and W6.
- A loop-back cable is connected to the DRV11–WA board.
- The DRV11–WA board is in a slot from 8 to 12. If the DRV11–WA is in a slot other than 8 to 12, you might get timeout errors.

When UETP testing is completed, restore the DRV11–WA to the proper operating configuration.

10.3.11 The DR750 or DR780

Note: Only Digital Customer Service personnel should set up the DR750 or DR780 for UETP testing.

To prepare the DR750 or the DR780 for UETP testing, use the following procedure:

1. Copy the DR780 microcode file, XF780.ULD, from the diagnostic medium to SYS$SYSTEM. Use the procedure described in the documentation provided with the DR780 Microcode Kit.
2. Turn off the power to the DR780.
3. Make the following DR780 backplane jumper changes:
   - Remove the jumper from W7 and W8.
   - Add a jumper from E04M1 to E04R1.
   - Add a jumper from E04M2 to E04R2.
4. Disconnect the DDI cable from the DR780. This cable is either a BC06V–nn cable, which can be disconnected, or a BC06R–nn cable, which requires that you remove its paddle card from the backplane of the DR780.
5. Restore power to the DR780.

When UETP testing is completed, restore the DR750 or the DR780 to the proper operating configuration.

10.3.12 The MA780

Be sure that the MA780 is set up according to the guidelines for shared memory in the *Guide to Maintaining a VMS System*.

If you run the MA780 device test individually, the logical name CTRLNAME must be defined as MPM, regardless of the memory name. As an alternative, you can enter ‘‘MPM’’ in response to the controller designation prompt.

10.3.13 A Second LPA11–K

If you have two LPA11–Ks, be sure that each is given a systemwide logical name in the SYS$MANAGER:LPA11STRT.COM file. The logical name for the first LPA11–K should be LPA11$0, and the logical name for the second LPA11–K should be LPA11$1.

10.3.14 Devices Not Tested

UETP does not test the following devices; their status has no effect on UETP execution:

- Devices that require operator interaction (such as card readers)
- Software devices (such as the null device and local memory mailboxes)

UETP does not have specific tests for UDA, HSC, or CI devices; they are tested implicitly by the disk, magnetic tape, and DECnet–VAX tests.

UETP also does not test the console terminal or console drives. If you boot the system, log in, and start UETP, you have shown that these devices can be used.

10.3.15 VAXcluster Testing

Before you run UETP in a VAXcluster environment, you should check the SYSTEST_CLIG account. The SYSTEST_CLIG account parallels SYSTEST except that it is dedicated to running the cluster-integration test. The requirements for the SYSTEST_CLIG account are as follows:

1 The account should be present in the user authorization file, exactly as distributed by Digital on each system in your VAXcluster.

Note: The SYSTEST_CLIG account might have been disabled as part of the VMS upgrade procedure. If it was disabled, you should re-enable the SYSTEST_CLIG account and give it a null password before you run UETP.

To re-enable the SYSTEST_CLIG account, enter the following commands and press RETURN after each one:

```
$   SET DEFAULT SYS$SYSTEM
$   RUN AUTHORIZE
UAF>   MODIFY /FLAGS=NODISUSER  /NOPASSWORD SYSTEST_CLIG
UAF>   EXIT
```

Note: Digital recommends that you disable the SYSTEST_CLIG account after testing has completed.

To disable the SYSTEST_CLIG account, enter the following commands and press RETURN after each one:

```
$   SET DEFAULT SYS$SYSTEM
$   RUN AUTHORIZE
UAF>   MODIFY /FLAGS=DISUSER SYSTEST_CLIG
UAF>   EXIT
```

2 The privileges and quotas of the SYSTEST_CLIG account should match those of the SYSTEST account.

UETP requires little additional preparation for the cluster-integration test phase beyond the requirements for other UETP test phases. The additional requirements for cluster integration testing are as follows:

1 Your system must be a member of a VAXcluster. If it is not, UETP displays a message and does not attempt to run the test.

2 Your system must use the same deadlock detection interval as the other systems in the VAXcluster.

3 The files UETCLIG00.COM and UETCLIG00.EXE, located in SYS$TEST, are necessary for each system included in the test.

4 DECnet–VAX must be set up between the VAXcluster nodes; UETP uses DECnet–VAX to create a process on those nodes. All checks that the test makes depend on its ability to create the SYSTEST_CLIG processes and to communicate with them using DECnet–VAX.

5 There must be a [SYSTEST] or [SYS0.SYSTEST] directory on some disk available to the VAXcluster for each node (both VMS and HSC) in the cluster. The test uses the same directory as the UETP disk test to create a file on each cluster node and to see if some other VMS node in the cluster can share access to that file. There must be one such directory per node; the test continues with the next cluster node once it has finished with a file.

10.3.16 A Small-Disk System

After you install the VMS operating system on a small system disk (for example, an RD53), you may not have the 1200 blocks of free disk space required to run UETP successfully. If you do not have 1200 free blocks on your system disk, use VMSTAILOR or DECW$TAILOR to remove some files from the system disk before you run UETP. For instructions on using VMSTAILOR and DECW$TAILOR see Chapter 12.

10.3.17 DECnet–VAX

The DECnet–VAX phase of UETP uses more system resources than other tests. You can, however, minimize disruptions to other users by running the test on the ‘‘least busy’’ node.

By default, the file UETDNET00.COM specifies the node from which the DECnet–VAX test will be run. To run the DECnet–VAX test on a different node, enter the following command before you invoke UETP:

```
$ DEFINE/GROUP UETP$NODE_ADDRESS node_address
```

This command equates the group logical name UETP$NODE_ADDRESS to the node address of the node in your area on which you want to run the DECnet–VAX phase of UETP.

For example:

```
$  DEFINE/GROUP UETP$NODE_ADDRESS 9.999
```

Note: When you use the logical name UETP$NODE_ADDRESS, UETP tests only the first active circuit found by NCP. Otherwise, UETP tests all active testable circuits.

When you run UETP, a router node attempts to establish a connection between your node and the node defined by UETP$NODE_ADDRESS. Occasionally, the connection between your node and the router node might be busy or non-existent. When this happens, the system displays the following error messages:

```
%NCP-F-CONNEC, Unable to connect to listener
-SYSTEM-F-REMRSRC, resources at the remote node were insufficient

%NCP-F-CONNEC, Unable to connect to listener
-SYSTEM-F-NOSUCHNODE, remote node is unknown
```

10.4 Preparing Vector Processors and the VAX Vector Instruction Emulation Facility (VVIEF)

No preparation is required to test vector processors. UETP automatically loads all installed and enabled vector processors during the load phase, and automatically tests all installed and enabled vector processors during the device test phase.

However, UETP cannot load the VAX Vector Instruction Emulation Facility (VVIEF) during the load phase, and will not automatically test VVIEF. To test VVIEF, you must do the following before running UETP:

1. Edit the file UETCONT00.DAT to add the following line:

   Y Y UETVECTOR.EXE "DEVICE_TEST"

2. Make sure VVIEF was activated when the system was booted. To determine if the VVIEF was activated, issue the following DCL commands:

   ```
   $ X = F$GETSYI("VECTOR_EMULATOR")
   $ SHOW SYMBOL X
   ```

   If the system displays a value of 1, VVIEF is loaded; if the system displays a value of 0, VVIEF is not loaded.

The VVIEF test can be executed as an individual test using the RUN command, as described in Section 10.7.2.2.

10.5 Starting UETP

When you have logged in and prepared the system and devices, you are ready to begin the test.

To start UETP, enter the following command and press RETURN:

```
$  @UETP
```

UETP displays the following prompt:

```
Run "ALL" UETP phases or a "SUBSET" [ALL]?
```

Throughout the startup dialog, brackets indicate the default value, which you can choose by pressing RETURN.

When running UETP for the first time, it is a good idea to choose the default value (ALL) and run all the phases. If you choose ALL, UETP displays three more questions, which are described in Section 10.5.2 through Section 10.5.4. If you want to run all the test phases, skip the next section.

10.5.1 Running a Subset of Phases

You can run a single phase by entering SUBSET or S in response to the following prompt:

```
Run "ALL" UETP phases or a "SUBSET" [ALL]?
```

If you enter S or SUBSET, UETP prompts you for the phase you want to run as follows:

```
You can choose one or more of the following phases:

        DEVICE, LOAD, DECNET, CLUSTER

Phases(s):
```

There is no default; enter one or more phase names from the list. Separate two or more phases with spaces or commas.

If your choice includes the LOAD phase, UETP displays the three prompts described in the next sections. To run the LOAD phase, refer to Section 10.5.3.

If you exclude the LOAD phase, UETP responds with only two prompts:

```
How many passes of UETP do you wish to run [1]?
Do you want Long or Short report format [Long]?
```

Section 10.5.2 and Section 10.5.4 discuss these questions. After you answer both questions, the phase you have selected runs to completion.

10.5.2 Single Run Versus Multiple Passes

If you specified the default ALL or a subset of phases at the last prompt, UETP displays the following message:

```
How many passes of UETP do you wish to run [1]?
```

You can repeat the test run as many times as you want. If you enter 1 in response to the prompt (or press RETURN for the default), UETP stops after completing a single run. If you specify a number greater than 1, UETP restarts itself until it completes the number of passes (runs) specified.

You can run UETP once to check that the system is working, or many times to evaluate the system's response to continuous use. For example, a field service technician who is interested only in verifying that a newly-installed system works might run UETP once or twice. A manufacturing technician might let the system run for several hours as part of the system integration and test.

When you specify multiple UETP runs, you might want to request a short console log (see Section 10.5.4). Make certain that all line printers and hardcopy terminals have enough paper, because each run requires two pages.

10.5.3 Defining User Load for Load Test

After you specify the number of passes, UETP prompts you as follows:

```
How many simulated user loads do you want [n]?
```

Note: UETP displays this prompt only if you choose to run the LOAD phase, either implicitly (by running all phases), or explicitly (by running a subset and specifying the LOAD phase).

The load test simulates a situation in which a number of users (detached processes) are competing for system resources. In response to this prompt, enter the number of users you want to simulate for this test. The number in brackets is the default value that UETP computed for your system. The default value depends on the amount of memory and the paging and swapping space that your system has.

Although the given default value is the best choice, you can increase or decrease the user load by entering your own response to the prompt. However, be aware that an increase might cause the test to fail because of insufficient resources.

If you want to see UETP display the user load equation as it runs, see Section 10.6.2.

10.5.4 Report Formats

The following prompt allows you to choose one of two console report formats:

```
Do you want Long or Short report format [Long]?
```

If you choose the long report format (the default), UETP sends all error messages as well as information on the beginning and end of all phases and tests to the console terminal. UETP records all its output in the UETP.LOG file, regardless of your response to this question.

In many cases, it may not be convenient to have UETP write the bulk of its output to the terminal. For example, if you run UETP from a hardcopy terminal, the output printing can slow the progress of the tests. This delay may not be a problem if you have requested only one run; however, you may prefer to use the short format if you intend to run multiple passes of UETP from a hardcopy terminal.

If you request the short format, UETP displays status information at the console, such as error messages and notifications of the beginning and end of each phase. This information enables you to determine whether UETP is proceeding normally. If the short console log indicates a problem, you can look at UETP.LOG for further information. UETP.LOG contains all the output generated by the various phases, as well as the status information displayed at the console.

After you choose the report format, UETP initiates its sequence of tests and runs to completion. If UETP does not complete successfully, refer to Section 10.6 for troubleshooting information.

10.5.5 Termination of UETP

At the end of a UETP pass, the master command procedure UETP.COM displays the time at which the pass ended. In addition, UETP.COM determines whether UETP needs to be restarted. (You can request multiple passes when you start up the test package; see Section 10.5.2.)

At the end of an entire UETP run, UETP.COM deletes temporary files and does other cleanup activities.

Pressing CTRL/Y or CTRL/C lets you terminate a UETP run before it completes normally. Normal completion of a UETP run, however, includes the deletion of miscellaneous files that have been created by UETP for the purpose of testing. The use of CTRL/Y or CTRL/C might interrupt or prevent these cleanup procedures.

The effect of these control characters depends on what part of UETP you are executing. For an explanation of the organization of UETP and its components, refer to Section 10.7.

10.5.5.1 Using CTRL/Y

Press CTRL/Y to abort a UETP run. Note, however, that cleanup of files and network processes in the [SYSTEST] directory may not be complete.

If you are running an individual test image, pressing CTRL/Y interrupts the current UETP test and temporarily returns control to the command interpreter. While the test is interrupted, you can enter a subset of DCL commands that are executed within the command interpreter and do not cause the current image to exit. The *VMS DCL Concepts Manual* contains a table of commands that you can use within the command interpreter. In addition, you can enter any of the following commands:

- The CONTINUE command continues the test from the point of interruption (except during execution of the cluster test).
- The STOP command terminates the test; the test aborts and control returns to the command interpreter.

Note: Using the STOP command may prevent cleanup procedures from executing normally. You should use the EXIT command if you want the image to do cleanup procedures before terminating.

- The EXIT command does cleanup procedures and terminates the test (except during execution of the cluster test); control returns to the command interpreter.

If you enter any DCL command other than CONTINUE, STOP and EXIT, the test does cleanup procedures and terminates, and the DCL command executes.

10.5.5.2 Using CTRL/C

Press CTRL/C to interrupt a UETP run. You cannot continue the same test phase after you press CTRL/C. UETP automatically goes to the next phase in the master command procedure.

Some UETP phases react to CTRL/C by cleaning up all activity and terminating immediately. Such tests display the following message:

```
%UETP-I-ABORTC, 'testname' to abort this test, type ^C
```

The phases that do not display the previous message terminate all processes they have started. These processes might not have a chance to complete normal cleanup procedures.

If you are running an individual test image, however, you can use CTRL/C to terminate the execution of the image and complete cleanup procedures.

Note that CTRL/C does not complete cleanup procedures for the cluster test.

10.6 Troubleshooting

This section explains the role of UETP in interpreting operational errors in a VMS operating system. Section 10.6.4 discusses common errors that can appear in a UETP run and describes how to correct them.

10.6.1 Relationship of UETP to Error Logging and Diagnostics

When UETP encounters an error, it reacts like a user program. It either returns an error message and continues, or it reports a fatal error and terminates the image or phase. In either case, UETP assumes the hardware is operating properly and it does not attempt to diagnose the error.

If the cause of an error is not readily apparent, use the following methods to diagnose the error:

- *VMS Error Log Utility*—Run the Error Log Utility to obtain a detailed report of hardware and system errors. Error log reports provide information about the state of the hardware device and I/O request at the time of each error. For information about running the Error Log Utility, refer to the *VMS Error Log Utility Manual*.
- *Diagnostic facilities*—Use the diagnostic facilities to test exhaustively a device or medium to isolate the source of the error.

10.6.2 Interpreting UETP Output

You can monitor the progress of UETP tests at the terminal from which they were started. This terminal always displays status information, such as messages that announce the beginning and end of each phase and messages that signal an error.

The tests send other types of output to various log files, depending on how you started the tests. The log files contain output generated by the actual test procedures. Even if UETP completes successfully, with no errors displayed at the terminal, it is good practice to check these log files for errors. Furthermore, when errors are displayed at the terminal, check the log files for more information about their origin and nature.

Each test returns a final completion status to the test controller image, UETPHAS00, using a termination mailbox. This completion status is an unsigned longword integer denoting a condition value. As a troubleshooting aid, UETPHAS00 displays the test's final completion status using the $FAO and $GETMSG system services. Sometimes, however, the $FA0 service needs additional information which cannot be provided using the termination mailbox. When this happens, UETP displays an error message similar to the following:

```
UETP-E-ABORT, !AS aborted at !%D
```

When UETP displays these types of error messages, check the log files for more information. You can also run the individual test to attempt to diagnose the problem.

The error messages that appear at the terminal and within the log files have two basic sources:

- UETP tests
- System components that are tested

To interpret the messages, you might need to refer either to the *VMS System Messages and Recovery Procedures Reference Manual* or to the manual that describes the individual system component.

Several parts of UETP, such as some device tests, UETINIT00.EXE, UETCLIG00.EXE, and UETDNET00.COM, let you obtain additional information concerning the progress of the test run or the problems it encounters. Because this information is usually insignificant, it is not displayed on the screen. To view the information, enter the following command and run the program:

```
$  DEFINE MODE DUMP
```

The following example shows the output for UETINIT00.EXE on a VAX 11/750:

```
$  RUN UETINIT00

        Welcome to VAX/VMS UETP Version V5.4

%UETP-I-ABORTC, UETINIT00 to abort this test, type ^C

You are running on an 11/750 CPU with 8704 pages of memory.
The system was booted from _DRA0:[SYS0.].
```

```
Run "ALL" UETP phases or a "SUBSET" [ALL]?
How many passes of UETP do you wish to run [1]?

The default number of loads is the minimum result of

1) CPU_SCALE * ((MEM_FREE + MEM_MODIFY) / (WS_SIZE * PER_WS_INUSE))
        0.80 * ((    8704 +        323) / (    350 *         0.20))  = 103

2) Free process slots                                                 = 56

3) Free page file pages / Typical use of page file pages per process
                  18040 /                                       1000 = 18

How many simulated user loads do you want [18]?
Do you want Long or Short report format [Long]?

UETP starting at  19-JUN-1990 09:08:26.71 with parameters:
DEVICE LOAD DECNET CLUSTER  phases, 1 pass, 18 loads, long report.
$
```

This program does not initiate any phase; it displays the equation used by UETP to determine user load and the specific factors that are employed in the current run.

You should respond to the questions by pressing RETURN. After you respond to the first prompt, the program displays the expressions that determine the default number of simultaneous processes. The following definitions apply:

- CPU_SCALE refers to the relative processing power of the CPU in relation to a VAX–11/780. For example, a VAX–11/785 has a CPU_SCALE of 1.5 because it has 1.5 times the processing power of a VAX–11/780 (1.0).
- MEM_FREE represents memory in pages available to users.
- MEM_MODIFY represents memory pages on the modified page list.
- WS_SIZE represents working set size.
- PER_WS_INUSE represents typical percentage of the working set in active use for each process.

UETINIT00 also displays the specific values represented by the expressions. In this example, UETP selects 18 as the default for simulated user loads, because 18 is the minimum result of the three expressions.

You should deassign the logical name MODE before running UETP, unless you prefer to see the previous breakdown every time you run UETP.

10.6.2.1 Defining a Remote Node for UETP Ethernet Testing

Occasionally during the UETUNAS00 test, it is difficult to determine whether the problem reports concern the device under test or the remote device. The easiest way to ensure proper error reporting is to define a ‘‘good turnaround.’’ A ‘‘good turnaround’’ is a remote node that you know turns around Ethernet packets correctly and is up and waiting in the ready state.

You can make the UETUNAS00 test use a known ‘‘good turnaround’’ by performing the following actions. In the commands that follow, assume that the ‘‘good’’ device is on node BETA, and that node BETA is already defined in the network database.

1 Find the address of the ‘‘good’’ Ethernet node by using the Network Control Program (NCP). In order to use NCP, the following conditions must apply:

- DECnet–VAX must be up and running on the system.
- The account you are using must have TMPMBX and NETMBX privileges.

Enter the following commands and press RETURN:

```
$  RUN SYS$SYSTEM:NCP
NCP>  TELL BETA SHOW CHARACTERISTICS ACTIVE LINES
```

If node BETA has not been defined in your network database, NCP displays an error message. In this event, specify another ‘‘good’’ node and retry the command. Otherwise, see your system or network manager.

NCP displays information similar to the following:

```
Active Line Volatile Characteristics as of  19-JUN-1990 16:13:02
                Line = UNA-0
                Counter timer            = 28800
                Receive buffers          = 6
                Controller               = normal
                Protocol                 = Ethernet
                Service timer            = 4000
                Hardware address         = AA-00-04-00-46-D3
                UNA device buffer size   = 1498
```

2 Use the displayed *hardware address* (in this case, AA00040046D3) to define the logical name TESTNIADR to point to the ‘‘good turnaround.’’ Note that you do *not* specify the hyphens (-).

First, log in to the SYSTEST account. Then enter the following command:

```
$  DEFINE/SYSTEM TESTNIADR AA00040046D3
```

3 Run UETP.

4 When UETP has completed, deassign the logical name TESTNIADR by entering the following command:

```
$  DEASSIGN/SYSTEM TESTNIADR
```

10.6.3 Log Files

At the end of a UETP run, the directory SYS$TEST contains a log file named UETP.LOG. This file contains all information generated by all UETP tests and phases. If the run involves multiple passes, you will find a version of UETP.LOG for each pass.

Although UETP.LOG contains information from all the passes, only information from the latest run is stored in this file. Information from the previous run is stored in a file named OLDUETP.LOG, which also has a version for each pass. Using these two files, UETP provides the output from its tests and phases from the two most recent runs.

The cluster test creates a NETSERVER.LOG file in SYS$TEST for each pass on each system included in the run. If the test is unable to report errors (for example, if the connection to another node is lost), the NETSERVER.LOG file on that node contains the result of the test run on that node. UETP does not purge or delete NETSERVER.LOG files; therefore, you must delete them occasionally to recover disk space.

If a UETP run does not complete normally, SYS$TEST might contain other log files. Ordinarily these log files are concatenated and placed within UETP.LOG. You can use any log files that appear on the system disk for error checking, but you must delete these log files before you run any new tests. You may delete these log files yourself or rerun the entire UETP, which checks for old UETP.LOG files and deletes them.

10.6.4 Possible UETP Errors

This section is intended to help you identify and solve problems you might encounter running UETP. You should refer to this section if you need help understanding a system failure and isolating its cause. This section is not intended as a repair manual and is not expected to diagnose any flaws in your system. It should, however, help you to interpret and act upon the information in the error messages.

If you are unable to correct an error after following the steps in this section, you should contact your Digital Customer Service representative. Any information you can supply about the measures you have taken to isolate the problem will help your Digital Customer Service representative diagnose the problem.

The following are the most common failures encountered while running UETP:

1. Wrong quotas, privileges, or account
2. UETINIT01 failure
3. Ethernet device allocated or in use by another application
4. Insufficient disk space
5. Incorrect VAXcluster setup
6. Problems during the load test
7. DECnet–VAX error
8. Lack of default access for the FAL object
9. Errors logged but not displayed
10. No PCB or swap slots
11. Hangs
12. Bugchecks and machine checks

The following sections describe these errors and offer the best course of action for dealing with each one.

10.6.4.1 Wrong Quotas, Privileges, or Account

If your assigned quotas or privileges do not match standard quotas and privileges for the SYSTEST account, UETP displays the following error message:

```
**********************
*  UETINIT00          *
*  Error count =  1   *
**********************
-UETP-W-TEXT,   The following:

        OPER privilege,
        BIOLM  quota,
        ENQLM  quota,
        FILLM  quota,

are nonstandard for the SYSTEST account and may result in UETP errors.
```

This message informs you that the OPER privilege and the BIOLM, ENQLM, and FILLM quotas either are not assigned correctly or are not assigned at all.

Note: UETP displays a similar message if you run the cluster integration test phase, and the privileges and quotas for the SYSTEST_CLIG account are incorrect. The SYSTEST and SYSTEST_CLIG accounts require the same privileges and quotas. Take the same action described in this section.

Solution

To correct the problem, use the following procedure:

1 Display all privileges and quotas in effect for the SYSTEST account using the AUTHORIZE utility as follows:

```
$  RUN SYS$SYSTEM:AUTHORIZE
UAF> SHOW SYSTEST

Username: SYSTEST                         Owner:  SYSTEST-UETP
Account:  SYSTEST                         UIC:    [1,7] ([SYSTEST])
CLI:      DCL                             Tables: DCLTABLES
Default:  SYS$SYSROOT:[SYSTEST]
LGICMD:   LOGIN
Login Flags:
Primary days:   Mon Tue Wed Thu Fri Sat Sun
Secondary days:
No access restrictions
Expiration:            (none)    Pwdminimum:  8   Login Fails:     0
Pwdlifetime:         14 00:00    Pwdchange:   19-JUN-1990 10:12
Last Login:            (none) (interactive),           (none) (non-interactive)
Maxjobs:         0  Fillm:       100  Bytlm:        32768
Maxacctjobs:     0  Shrfillm:      0  Pbytlm:           0
Maxdetach:       0  BIOlm:       300  JTquota:       1024
Prclm:           8  DIOlm:       300  WSdef:          256
Prio:            4  ASTlm:       100  WSquo:          512
Queprio:         0  TQElm:       100  WSextent:      2048
CPU:        (none)  Enqlm:       300  Pgflquo:      45000
Authorized Privileges:
  CMKRNL CMEXEC SYSNAM GRPNAM DETACH DIAGNOSE LOG_IO GROUP
  PRMCEB PRMMBX SETPRV TMPMBX NETMBX VOLPRO PHY_IO SYSPRV
Default Privileges:
  CMKRNL CMEXEC SYSNAM GRPNAM DETACH DIAGNOSE LOG_IO GROUP
  PRMCEB PRMMBX SETPRV TMPMBX NETMBX VOLPRO PHY_IO SYSPRV
UAF>  SHOW SYSTEST_CLIG
   .
   .
   .
UAF>  EXIT
```

2 Check that the privileges and quotas assigned to the account match the following:

Privileges

CMKRNL	CMEXEC	NETMBX	DIAGNOSE
DETACH	PRMCEB	PRMMBX	PHY_IO
GRPNAM	TMPMBX	VOLPRO	LOG_IO
SYSNAM	SYSPRV	SETPRV	GROUP

Quotas

BIOLM: 18	PRCLM: 8
DIOLM: 55	ASTLM: 100
FILLM: 100	BYTLM: 32768
TQELM: 20	CPU: no limit
ENQLM: 300	PGFLQUOTA: 20480
WSDEFAULT: 256	WSQUOTA: 512
WSEXTENT: 2048	

3 If any privileges or quotas are incorrect, run the Authorize Utility (AUTHORIZE) to add them (AUTHORIZE is explained in the *VMS Authorize Utility Manual*). As an alternative, you can temporarily assign the correct privileges with the DCL command SET PROCESS/PRIVILEGES.

If you are logged in to the wrong account, the following error message asks you to log in to the SYSTEST account:

```
$  @UETP

**********************
*  UETINIT00         *
*  Error count =  1  *
**********************
-UETP-E-ABORT, UETINIT00 aborted at  19-JUN-1990 14:24:10.13
-UETP-E-TEXT, You are logged in to the wrong account.
              Please log in to the SYSTEST account.
$
```

You must run UETP from the SYSTEST account.

10.6.4.2 UETINIT01 Failure

UETINIT01 failures are related to peripheral devices; this type of error message might indicate any of the following:

- Device failure
- Device not supported or not mounted
- Device allocated to another user
- Device write-locked

- Lost vacuum on a magnetic tape drive
- Drive off line

In some cases, the course of action you should take is specified explicitly in the error message. For example, you might receive a message from the Operator Communication Facility (OPCOM) process informing you of a problem and recommending a corrective measure:

```
%OPCOM,  19-JUN-1990 14:10:52.96, request 1, from user SYSTEST
Please mount volume UETP in device _MTA0:
%MOUNT-I-OPRQST, Please mount volume UETP in device _MTA0:
```

Other error messages might relate information in which the solution is specified implicitly:

```
%UETP-S-BEGIN, UETDISK00 beginning at 19-JUN-1990 13:34:46.03

**********************
*  DISK_DRA          *
*  Error count =  1  *
**********************
-UETP-E-TEXT, RMS file error in file DRA0:DRA00.TST
-RMS-E-DNR, device not ready or not mounted
%UETP-S-ENDED, UETDISK00 ended at  19-JUN-1990 13:34:46.80
```

This message tells you that a disk drive is either not ready or not mounted. From this information, you know where to look for the cause of the failure—at the disk drive. If you cannot see the cause of the problem immediately, check the setup instructions in Section 10.3.

In other cases, the cause of a failure might not be obvious from the information in the message. The problem might be related to hardware rather than software. For example, the Ethernet adapter test may produce one of the following messages if UETP does not have exclusive access to the Ethernet adapter:

- Inter-module cable unplugged
- Self-test failure code 0000000

To run the self-test diagnostic on the Ethernet adapter successfully, UETP needs exclusive access to the adapter. As explained in Section 10.3.8, you must shut down DECnet–VAX and the LAT terminal server before running the UETP device test phase if you want to test the Ethernet adapter.

Solution

To determine where or when the failure occurs in the execution of UETP, use the following procedure:

- Run the device test individually (see Section 10.5.1). By doing this, you can determine if the failure can be re-created. Also, you are isolating the cause of the problem by reproducing it using the least amount of software possible. For example, if the failure occurs only when you run the entire device phase, and not when you run the affected device test individually, you can conclude the problem is related to device-interaction. Conversely, if you can re-create the error by running the single device test, then you have proved that the error is not related to device interaction.

- Run the device test with different media. If your run of the single device test succeeded in reproducing the error, the magnetic tape or disk media could be defective. Running the same test with new media determines whether this is the problem.
- Call Digital Customer Service. If you have tried all the previous steps without solving the problem, you should contact your Digital Customer Service representative.

10.6.4.3 Device Allocated or in Use by Another Application

If DECnet–VAX or the LAT software is running during the DEVICE phase, the UETUNAS00 test displays the following message:

```
-UETP-W-TEXT, Device is in use by DECnet or another application
```

Other UETP communication device tests display the following message:

```
SYSTEM-W-DEVALLOC, device already allocated to another user
```

Solution

If you want to run the device test on the Ethernet Adapter, shut down DECnet–VAX and LAT software before beginning the test.

10.6.4.4 Insufficient Disk Space

When you run continuous passes of UETP, log files accumulate on the disk from which UETP was run. These files reduce the amount of free disk space available for each successive pass. If the amount of disk space available becomes too small for the current load, the following error message appears:

```
%UETP-S-BEGIN, UETDISK00 beginning at  19-JUN-1990 08:12:24.34
%UETP-I-ABORTC, DISK_DJA to abort this test, type ^C

**********************
*  DISK_DJA          *
*  Error count = 1   *
**********************
-UETP-F-TEXT, RMS file error in file DJA0:DJA00.TST
-RMS-F-FUL, device full (insufficient space for allocation)

**********************
*  DISK_DJA          *
*  Error count = 2   *
**********************
-UETP-F-TEXT, RMS file error in file DJA0:DJA01.TST
-RMS-F-FUL, device full (insufficient space for allocation)
%UETP-E-DESTP, DISK_DJA stopped testing DJA unit 0 at 08:12:36.91
%UETP-S-ENDED, UETDISK00 ended at  19-JUN-1990 08:12:37.98
```

Solution

Make more space available on the disk. You can do this by using one or more of the following techniques:

- Delete unnecessary files to create more space.
- Purge files, if multiple versions exist.
- Mount a volume with sufficient space.

- Check for disk quotas that may be enabled on the disk. If disk quotas are enabled, either disable or increase them (see the *VMS SYSMAN Utility Manual* for a description of the Disk Quota Utility).
- Run VMSTAILOR if you have a small-disk system. See Chapter 4 for more information.

See Section 10.2.1 and Section 10.3.2 for a further discussion of disk space.

10.6.4.5 Incorrect Setup of a VAXcluster

Most problems that can occur during the cluster-integration test are related to improper setup of the VAXcluster or of UETP on the VAXcluster. These problems are most likely to occur at the following stages of the VAXcluster test:

- Near the beginning, when processes on VMS nodes are started
- Toward the end, when cluster file access is checked

The cluster test phase shows that various VMS nodes in your cluster can simultaneously access files on selected nodes in the cluster. First, UETP tries to create a file on a disk drive that is accessible to the other selected nodes in the cluster. The following are the requirements for creating a file in the cluster test phase:

- There must be a [SYSTEST] directory on the disk in either the master file directory (MFD) or in the root directory [SYS0.].
- The [SYSTEST] directory must be protected so that the SYSTEST account can create a file in it.

If UETP is unable to find a suitable device on a certain node, the test displays a warning message and proceeds to the next cluster node.

Nodes on which the operator's terminal (OPA0) is set to the ''No broadcast'' terminal characteristic will generate the following error message during the cluster test:

```
**********************
*  UETCLIG00master   *
*  Error count =  1  *
**********************
-UETP-E-TEXT, 0 operator consoles timed out on the cluster test warning
       and 1 operator console rejected it.
-UETP-E-TEXT, Status returned was,
      "%SYSTEM-F-DEVOFFLINE, device is not in configuration or not
      available"
```

Disregard this message if OPA0 is set to ''No broadcast.''

Solution

Whenever you suspect a problem, you should try to recover the SYS$TEST:NETSERVER.LOG file that was created when the SYSTEST_CLIG process was created. This file may contain additional error information that could not be transmitted to the node running the test. If it was not possible to create the SYSTEST_CLIG process on some node, the system accounting file for that node may contain a final process status in a process termination record.

The following problems can occur during a cluster test:

- Logging in at other nodes—This problem is due to incorrect setup for the cluster test at the remote VMS node. For example, if you specified a password for the SYSTEST_CLIG account or if you disabled the SYSTEST_CLIG account, the test displays the following message:

  ```
  %SYSTEM-F-INVLOGIN, login information invalid at remote node
  ```

  Refer to Section 10.3.15 and Section 10.6.2.1 for information on preparing for VAXcluster testing.

- Communicating with other nodes—A message indicates a DECnet–VAX problem. Check the NETSERVER.LOG file on the affected node to determine the cause.

- Taking out locks or detecting deadlocks—The most likely cause of this problem is that you are not logged in to the SYSTEST account. Another possibility is that your cluster is not configured properly.

- Creating files on VAXcluster nodes—This problem is due to incorrect setup for the cluster test; refer to Section 10.3.15 for information on preparing for VAXcluster testing.

10.6.4.6 Problems During the Load Test

A variety of errors can occur during the load test because the command procedures that are started during the tests run several utilities and do many functions. Tracking a problem can be difficult because UETP deletes the log files that are generated during the load test (see Section 10.7.3).

Solution

If a problem occurs during the load test and the cause is not obvious, you can modify UETP.COM to preserve the log files as follows:

1 Add the /NODELETE qualifier to the following line:

```
$ TCNTRL UETLOAD00.DAT/PARALLEL_COUNT='LOADS/REPORT_TYPE='REPORT
```

2 Delete the following line:

```
$ DELETE UETLO*.LOG;*
```

Rerun the load test with these changes to try to re-create the problem.

If you re-create the problem, look at the contents of the appropriate log file. You can determine which log file to read by understanding the scheme by which the load test names its processes and log files. (The log file names are derived from the process names.)

The load test creates processes that are named in the following format:

UETLOADnn_nnnn

For example:

```
%UETP-I-BEGIN, UETLOAD00 beginning at 19-JUN-1990 15:45:08.97
%UETP-I-BEGIN, UETLOAD02_0000 beginning at 19-JUN-1990 15:45:09.42
%UETP-I-BEGIN, UETLOAD03_0001 beginning at 19-JUN-1990 15:45:09.63
%UETP-I-BEGIN, UETLOAD04_0002 beginning at 19-JUN-1990 15:45:10.76
%UETP-I-BEGIN, UETLOAD05_0003 beginning at 19-JUN-1990 15:45:11.28
%UETP-I-BEGIN, UETLOAD06_0004 beginning at 19-JUN-1990 15:45:12.56
%UETP-I-BEGIN, UETLOAD07_0005 beginning at 19-JUN-1990 15:45:13.81
%UETP-I-BEGIN, UETLOAD08_0006 beginning at 19-JUN-1990 15:45:14.94
%UETP-I-BEGIN, UETLOAD09_0007 beginning at 19-JUN-1990 15:45:16.99
%UETP-I-BEGIN, UETLOAD10_0008 beginning at 19-JUN-1990 15:45:19.32
%UETP-I-BEGIN, UETLOAD11_0009 beginning at 19-JUN-1990 15:45:19.94
%UETP-I-BEGIN, UETLOAD02_0010 beginning at 19-JUN-1990 15:45:20.20
%UETP-I-BEGIN, UETLOAD03_0011 beginning at 19-JUN-1990 15:45:21.94
%UETP-I-BEGIN, UETLOAD04_0012 beginning at 19-JUN-1990 15:45:22.99
```

Note that if more than 10 processes are created, the numbering sequence for the UETLOADnn portion of the process name starts over at UETLOAD02; however, the 4 digits of the _nnnn portion continue to increase.

Each load test process creates two log files. The first log file is created by the test controller; the second log file is created by the process itself. The log file that you need to look at for error information on any given load test process is the one that was created by the test controller (the first log file).

The load test log file derives its file name from the process name, appending the last four digits of the process name (from the _nnnn portion) to UETLO. The test-controller log file and the process log file for each process use the same file name; however, the process log file has the higher version number of the two. For example, the log files created by the process UETLOAD05_0003 would be named as follows:

UETLO0003.LOG;1 (test-controller log file)

UETLO0003.LOG;2 (process log file)

Make sure that you look at the log file with the lower version number; that file contains the load test commands and error information.

After you have isolated the problem, restore UETP.COM to its original state and delete the log files from the load test (UETLO*.LOG;*); failure to delete these files might result in disk space problems.

10.6.4.7 DECnet–VAX Error

A DECnet–VAX error message might indicate that the network is unavailable.

Solution

- If DECnet–VAX is included in your system, register the product authorization key (PAK). For instructions, see the *VMS Version 5.4 Release Notes*.
- If DECnet–VAX is not included in your system, ignore the message; it is normal and does not affect the UETP run.

If you encounter other DECnet–VAX-related errors, you should do the following:

- Run DECnet–VAX as a single phase (see Section 10.5.1) to determine whether the error can be re-created.
- Refer to the *VMS System Messages and Recovery Procedures Reference Manual*.

10.6.4.8 Errors Logged but Not Displayed

If no errors are displayed at the console terminal or reported in the UETP.LOG file, you should run the Error Log Utility to see if any errors were logged in the ERRLOG.SYS file. See the *VMS Error Log Utility Manual* for information about running the Error Log Utility.

10.6.4.9 No PCB or Swap Slots

The following error message indicates that no process control block (PCB) or swap slots are available:

```
%UETP-I-BEGIN, UETLOAD00 beginning at  19-JUN-1990 07:47:16.50
%UETP-I-BEGIN, UETLOAD02_0000 beginning at  19-JUN-1990 07:47:16.76
%UETP-I-BEGIN, UETLOAD03_0001 beginning at  19-JUN-1990 07:47:16.92
%UETP-I-BEGIN, UETLOAD04_0002 beginning at  19-JUN-1990 07:47:17.13
%UETP-I-BEGIN, UETLOAD05_0003 beginning at  19-JUN-1990 07:47:17.35
%UETP-I-BEGIN, UETLOAD06_0004 beginning at  19-JUN-1990 07:47:17.61
%UETP-W-TEXT, The process -UETLOAD07_0005- was unable to be created,
  the error message is
-SYSTEM-F-NOSLOT, no pcb or swap slot available
%UETP-W-TEXT, The process -UETLOAD08_0006- was unable to be created,
  the error message is
-SYSTEM-F-NOSLOT, no pcb or swap slot available
%UETP-W-TEXT, The process -UETLOAD09_0007- was unable to be created,
  the error message is
-SYSTEM-F-NOSLOT, no pcb or swap slot available
%UETP-W-TEXT, The process -UETLOAD10_0008- was unable to be created,
  the error message is
-SYSTEM-F-NOSLOT, no pcb or swap slot available
%UETP-W-TEXT, The process -UETLOAD11_0009- was unable to be created,
  the error message is
-SYSTEM-F-NOSLOT, no pcb or swap slot available
%UETP-W-ABORT, UETLOAD00 aborted at  19-JUN-1990 07:47:54.10
-UETP-W-TEXT, Aborted via a user CTRL/C.
 ************************************************
 *                                              *
    END OF UETP PASS 1 AT  19-JUN-1990 07:48:03.17
 *                                              *
 ************************************************
```

Solution

To solve this problem, use the following procedure:

1. Rerun individually the phase that caused the error message (the LOAD phase in the previous example) to see if the error can be reproduced.
2. Increase the size of the page file, using either the command procedure SYS$UPDATE:SWAPFILES.COM or the System Generation Utility (see the *VMS System Generation Utility Manual*).

3 Increase the SYSGEN parameter MAXPROCESSCNT, if necessary, and reboot the system.

4 Increase both the page file size and the MAXPROCESSCNT, if necessary.

10.6.4.10 Hangs

If there is no keyboard response or system disk activity, the system may be hung.

Solution

A system hang can be difficult to trace; you should always save the dump file for reference. To learn why the system hung, run the System Dump Analyzer as described in the *VMS System Dump Analyzer Utility Manual*. Reasons for a system hang include the following:

- Insufficient pool space—Increase the value of the SYSGEN parameter NPAGEVIR and reboot the system.
- Insufficient page file space—Increase the page file space using the System Generation Utility as described in the *VMS System Generation Utility Manual*.
- I/O device failure causing driver-permanent loop—Call Digital Customer Service.

10.6.4.11 Lack of Default Access for the FAL Object

With VMS Version 5.2, the network configuration procedure NETCONFIG.COM no longer provides access for the FAL network object by default. UETP, however, still assumes the existence of default access for the FAL object (this assumption will be removed in a future release of VMS). When you install DECnet using the defaults presented by the procedure NETCONFIG.COM, the UETP DECnet phase may produce error messages that were not seen in earlier versions.

If default FAL access is disabled at the remote node selected by UETP for DECnet testing (the adjacent node on each active circuit, or a node defined by the group logical name UETP$NODE_ADDRESS), messages similar to the following will appear:

```
%UETP-W-TEXT, The process -SVA019841_0001- returned a final status of:
%COPY-E-OPENOUT, error opening !AS as output
```

These messages are followed by:

```
%COPY-E-OPENOUT, error opening 9999"":SVA019841.D1; as output
-RMS-E-CRE, ACP file create failed
-SYSTEM-F-INVLOGIN, login information invalid at remote node
%COPY-W-NOTCOPIED, SYS$COMMON:[SYSTEST]UETP.COM;2 not copied
%UETP-E-TEXT, Remote file test data error
```

These messages can be ignored.

10.6.4.12 Bugchecks and Machine Checks

When the system aborts its run, a bugcheck message appears at the console.

Solution

Call Digital Customer Service. Often a hardware problem causes bugchecks and machine checks; there is no easy way to solve bugchecks or machine checks. It is important, however, that you save the SYS$SYSTEM:SYSDUMP.DMP and ERRLOG.SYS files so that they are available for examination. It is also important to know whether the failure can be re-created; you can verify this by running UETP again.

10.7 UETP Tests and Phases

This section explains in detail the organization of UETP and the individual components within the test package. You run UETP by starting a master command procedure containing commands to start each test phase. The procedure begins by prompting you for information needed by the various test phases. (See Section 10.5 for a detailed description of starting UETP.)

The master command procedure, UETP.COM, contains commands that initiate each test phase. UETP.COM also contains commands that do such tasks as defining logical names and manipulating files generated by the tests.

The UETP.COM procedure also issues commands to start the test controlling program, UETPHAS00.EXE, which, in turn, controls each test phase. The test controller starts up multiple detached processes. It also reports their completion status and other information the processes report to it.

The following sections describe the various UETP test phases.

10.7.1 Initialization Phase

The following occurs during the initialization phase:

- The image UETINIT00.EXE prompts you for information (see Section 10.5). Your information defines variables that affect the execution of UETP tests.
- The image UETINIT01.EXE gathers information on all the controllers in the system and on their associated devices. This image writes the information into a file called UETINIDEV.DAT.
- Using the information in UETSUPDEV.DAT, UETINIT01.EXE verifies which devices in the system are operable by running the appropriate device test. Each device test completes a simple read/write operation to each device. If a device fails this test, the device's entry in UETINIDEV.DAT specifies that the device cannot be tested. As a result, subsequent UETP tests ignore that device.

- For each testable controller, UETINIT01.EXE writes a line into a file called UETCONT00.DAT. The line associates a test file with the controller it tests.

A summary of UETINIDEV.DAT always exists in UETP.LOG, and UETINIT01.EXE sends that summary to the console if you have requested the long report format.

10.7.2 Device Test Phase

The device test phase includes separate tests for each type of device, such as disk, magnetic tape, line printer, and terminal. This section explains the device test phase and presents instructions for testing a single device. If you want to run the entire device test phase individually, refer to Section 10.5.1.

10.7.2.1 How the Device Phase Works

The UETP device test phase starts an executable image, the phase controller UETPHAS00, which creates a detached process for every device controller to be tested. For example, if a system includes three terminal controllers, one line printer controller, and two disk controllers, the image creates six detached processes. In parallel, the detached processes execute images that test the various types of devices.

The initialization phase of UETP creates a file called UETINIDEV.DAT and a file called UETCONT00.DAT. UETINIDEV.DAT contains data on the VMS-supported controllers in the system and their associated devices; UETCONT00.DAT associates a device test image with each testable controller.

UETPHAS00 uses the information in UETCONT00.DAT to find a device controller name to pass to each detached process that it creates. UETPHAS00 passes the controller name by writing it to a mailbox that is SYS$INPUT to individual tests. Each detached process uses that data to determine which controller to test. The test image then searches UETINIDEV.DAT for the device controller and for all testable units on that controller. The phase controller terminates when all devices on all controllers have completed testing.

Because UETCONT00.DAT is deleted automatically at the end of a UETP run, you cannot run the device phase unless you start UETP.COM; you can run only individual test images. UETINIDEV.DAT exists in SYS$TEST unless you explicitly delete it.

10.7.2.2 Running a Single Device Test

You must be logged in to the SYSTEST account to run the individual tests as described in this section. Also, a copy of UETINIDEV.DAT must exist. If a copy of the file is not present from a previous run (a run of the entire UETP or a run of the device test phase creates UETINIDEV.DAT), you can create it. Note that when you run a single test, no log file is created; the test sends all its output to your terminal.

If you do not want to test all the device types, you can test a specific controller by choosing a test image name from Table 10–1 and executing it as in the following example:

```
$  RUN UETTTYS00
Controller designation?:  TTB
```

UETP prompts you for the controller designation and the device code. Unless you are testing your own terminal, you must explicitly designate a controller name. If you are running the terminal test, you can press RETURN to test your terminal only.

If you plan to repeat the run several times, you might find it more convenient to define the logical name CTRLNAME as follows:

```
$  DEFINE CTRLNAME TTB
$  RUN UETTTYS00
```

When you define the controller name in this way, the logical name CTRLNAME remains assigned after the test completes. To deassign this logical name, use the DCL command DEASSIGN as follows:

```
$  DEASSIGN CTRLNAME
```

Format of UETINIDEV.DAT

The UETINIDEV.DAT file is an ASCII sequential file that you can type or edit if necessary. The contents of this file are shown in the following command sequence:

```
$  TYPE UETINIDEV.DAT

DDB x ddd
UCB y uuuuu nnnnnnnnn.nnn
END OF UETINIDEV.DAT
```

The symbols in this example are defined as follows:

Symbol	Value
x	T, if there are any testable units for this controller; N, if this controller is not to be tested
y	T, if this unit is testable; N, if this unit is not testable
ddd	Device controller name, for example DUA
uuuuu	Device unit number, for example 25
nnnnnnnnn.nnn	UETP device test name for the unit, for example, UETDISK00.EXE

UETINIDEV.DAT contains a DDB (device data block) line for each controller connected or visible to your system. After the DDB line there is a UCB (unit control block) line for each unit connected to that controller. In addition, if your system uses MA780 memory in a loosely coupled CPU configuration, UETINIDEV.DAT includes one UCB line for each MA780 memory. A device test can test a particular device only if both the DDB line and the UCB line indicate that the device is testable.

Running a Test in Loop Mode

If you want to put extra stress on a device, you can run the device test in loop mode, which causes the test to run indefinitely. For example:

```
$  DEFINE MODE LOOP
$  RUN UETDISK00
Controller designation?:  DRA
%UETP-I-TEXT, End of pass 1 with 980 iterations at 19-JUN-1990 16:18:51:03

^C
```

You must use CTRL/C to terminate the test run. If you use CTRL/Y, UETP does not complete cleanup procedures.

Functions of Individual Device Tests

For each disk in the system, the disk test allocates two files into which it randomly writes blocks of data. The test then checks the data, reports any errors to SYS$OUTPUT, and deletes the disk files.

When you run the disk test phase in a cluster environment, the test accesses all disks that are mounted by the system being tested, and users of the disk being tested might encounter an insufficient disk space problem. You should warn users on remote nodes (who share disks with users on the local system) that UETP may be testing a disk they are using.

The magnetic tape test exercises all the magnetic tape drives in the system. The test creates a large file on each mounted magnetic tape, into which it writes multiple sequential records of varying sizes. After writing the records, the test rewinds the magnetic tape, validates the written records, and re-initializes the magnetic tape.

The terminal and line printer test generates several pages or screens of output, in which each page or screen contains a header line and a test pattern of ASCII characters. A header line contains the test name, the device name, the date, and the time.

For the laboratory peripheral accelerator (LPA11–K), the test image determines the configuration on the LPA11–K's I/O bus. The image loads all types of microcode to the LPA11–K and reads or writes data for each device on the LPA11–K I/O bus.

The communications device tests fill the transmit message buffer with random data; then, using loopback mode, the tests transmit and receive the message several times. To check that the looped-back data is correct, an AST routine is associated with a $QIO read to compare the received message against the transmitted message. The procedure is repeated using messages of different lengths.

The interface device tests put the devices they are testing in maintenance mode, write random data, and then verify the data.

The MA780 device test creates and modifies mailboxes, common event flags, and global sections in shared memory; then it verifies that modifications can be made. You can run MA780 tests in parallel from separate systems so that the tests interact with each other through common MA780 memories.

The Ethernet adapter test does self-test diagnostics on the device. It also does read and write tasks with test data that uses various adapter modes (such as internal loopback and external loopback).

The vector processor device test performs simple vector-scalar and vector-vector arithmetic operations and compares the results with expected values. The test also uses vector-related system service extensions and forces the system to generate arithmetic and memory management exceptions.

Table 10–1 lists the device test images and the devices to be tested.

Table 10–1 Device Tests

Test Image Name	Devices Tested
UETDISK00.EXE	Disks
UETTAPE00.EXE	Magnetic tape drives and tape cartridge drives
UETTTYS00.EXE	Terminals and line printers
UETLPAK00.EXE	LPA11–K
UETCOMS00.EXE	DMC11, DMR11
UETDMPF00.EXE	DMF32, DMP11
UETDR1W00.EXE	DR11–W
UETDR7800.EXE	DR780, DR750
UETMA7800.EXE	MA780
UETCDRO00.EXE	RRD40, RRD50
UETUNAS00.EXE	Ethernet Adapters
UETVECTOR.EXE	Vector Processor, VVIEF

10.7.3 System Load Test Phase

The purpose of the system load test is to simulate a number of terminal users who are demanding system resources simultaneously. The system load tests, directed by the file UETLOAD00.DAT, create a number of detached processes that execute various command procedures. Each process simulates a user logged in at a terminal; the commands within each procedure are the same types of commands that a user enters from a terminal. The load test creates the detached processes in quick succession, and the processes generally execute their command procedures simultaneously. The effect on the system is analogous to an equal number of users concurrently issuing commands from terminals. In this way, the load test creates an environment that is similar to normal system use.

The load test uses the logical name LOADS to determine the number of detached processes to create. When you initiate the UETP command procedure, it prompts for the number of users to be simulated (see Section 10.5.3) and consequently the number of detached processes to be created. Your response, which depends on the amount of memory and the swapping and paging space in your system, defines the group logical name LOADS.

The UETP master command procedure deassigns all group logical names assigned by its tests as part of the termination phase. The group logical name LOADS remains assigned only if the UETP package does not complete normally.

The command procedures executed by the load test can generate a large amount of output, depending on the number of detached processes created. For each detached process (or user), the test creates a version of an output file called UETLOnnnn.LOG (''nnnn'' represents a string of numeric characters). The console displays only status information as the load test progresses.

Whether the load test runs as part of the entire UETP or as an individual phase, UETP combines the UETLOnnnn.LOG files, writes the output to the file UETP.LOG, and deletes the individual output files.

You can run the system load test as a single phase by selecting LOAD from the choices offered in the startup dialog (see Section 10.5.1).

10.7.4 DECnet–VAX Test Phase

If DECnet–VAX is included in your VMS system, a run of the entire UETP automatically tests DECnet–VAX hardware and software. Because communications devices are allocated to DECnet–VAX and the DECnet–VAX devices cannot be tested by the UETP device test, UETP will not test the Ethernet adapter if DECnet–VAX or another application has allocated the device. The DECnet–VAX node and circuit counters are zeroed at the beginning of the DECnet–VAX test to allow for failure monitoring during the run.

As with other UETP phases, you can run the DECnet–VAX phase individually by following the procedure described in Section 10.5.1.

10.7.4.1 Environment

The DECnet–VAX test will work successfully on VMS systems connected to all DECnet–VAX-supported node types, including routing and nonrouting nodes and several different types of operating systems (such as RSTS, RSX, TOPS, and RT). There must be some sort of default access on remote systems to copy files between systems. The DECnet–VAX phase tests the following:

- The node UETP is running on.
- All circuits in sequence, unless you have defined the logical name UETP$NODE_ADDRESS to be the remote node that you want to run the test on. If you have defined a remote node, the DECnet–VAX phase tests only one circuit.
- All adjacent or first-hop nodes and all circuits in parallel.

There is no limit on the number of communication lines supported by the tests. A test on one adjacent node should last no more than two minutes at normal communications transfer rates.

Note: UETP assumes your system has default access for the FAL object, even though the network configuration command procedure

NETCONFIG.COM does not provide access for the FAL object by default. When you install DECnet with the defaults presented by NETCONFIG.COM, the UETP DECnet phase might produce error messages. See Section 10.6.4.11 for more information.

10.7.4.2 How the DECnet–VAX Phase Works

UETP (under the control of UETPHAS00.EXE) reads the file UETDNET00.DAT and completes the following steps during the DECnet–VAX phase:

1. Executes a set of Network Control Program (NCP) LOOP EXECUTOR commands to test the node on which UETP is running.
2. Uses NCP to execute the command SHOW ACTIVE CIRCUITS. The results are placed in UETININET.TMP, from which UETP creates the data file UETININET.DAT. The UETININET.TMP file contains the following information for any circuit in the ON state but not in transition:
   - Circuit name
   - Node address
   - Node name (if one exists)

   The UETININET.TMP file is used throughout the DECnet–VAX phase to determine which devices to test.
3. Uses the UETININET.TMP file to create an NCP command procedure for each testable circuit. Each command procedure contains a set of NCP commands to zero the circuit and node counters and to test the circuit and adjacent node by copying files back and forth.

Note: If you do not want the counters zeroed, do not test DECnet–VAX.

4. Executes the command procedures from step 3 in parallel to simulate a heavy user load. The simulated user load is the lesser of the following values:
   - The number of testable circuits, multiplied by two
   - The maximum number of user-detached processes that can be created on the system before it runs out of resources (determined by UETINIT00)
5. Executes a program, UETNETS00.EXE, that uses the UETININET.DAT file to check the circuit and node counters for each testable circuit. If a counter indicates possible degradation (by being nonzero), its name and value are reported to the console. All counters are reported in the log file, but only the counters that indicate degradation are reported to the console. Following is an example of UETNETS00 output:

```
%UETP-S-BEGIN, UETNETS00 beginning at  19-JUN-1990 13:45:33.18
%UETP-W-TEXT, Circuit DMC-0 to (NODENAME1) OK.
%UETP-I-TEXT, Node  (NODENAME2) over DMC-1 response timeouts = 1.
%UETP-I-TEXT, Circuit DMC-1 to (NODENAME2) local buffer errors = 34.
%UETP-I-TEXT, Node  (NODENAME3) over DMP-0 response timeouts = 3.
%UETP-S-ENDED, UETNETS00 ended at  19-JUN-1990 13:45:36.34
```

Because degradation is not necessarily an error, the test's success is determined by you, not the system. The following counters indicate possible degradation:

For Circuits

- Arriving congestion loss
- Corruption loss
- Transit congestion loss
- Line down
- Initialization failure
- Data errors inbound
- Data errors outbound
- Remote reply timeouts
- Local reply timeouts
- Remote buffer errors
- Local buffer errors
- Selection timeouts
- Remote process errors
- Local process errors
- Locally initiated resets
- Network initiated resets

For Nodes

- Response timeouts
- Received connect resource errors
- Aged packet loss
- Node unreachable packet loss
- Node out of range packet loss
- Oversized packet loss
- Packet format error
- Partial routing update loss
- Verification reject

10.7.5 Cluster-Integration Test Phase

The cluster-integration test phase consists of a single program and a command file that depend heavily on DECnet–VAX. This phase uses DECnet–VAX to create SYSTEST_CLIG processes on each VMS node in the cluster and to communicate with each node. SYSTEST_CLIG is an account that is parallel to SYSTEST, but limited so that it can only be used as part of the cluster-integration test. The following restrictions on the SYSTEST_CLIG account are necessary for a correct run of the cluster test phase:

- The account must be enabled and the password must be null. For more information, see Section 10.3.15.
- The UIC must be the same as that of the SYSTEST account.
- The account must have the same privileges and quotas as the SYSTEST account. For more information, see Section 10.6.4.1.
- The account can allow login only through DECnet–VAX.
- The account must be locked into running UETCLIG00.COM when it logs in.

These items are necessary to ensure the security and privacy of your system. If the test cannot create a SYSTEST_CLIG process on some VMS node, it gives the reason for the failure and ignores that node for the lock tests and for sharing access during the file test. Also, the test does not copy log files from any node on which it could not create the SYSTEST_CLIG process. If there is a problem communicating with a SYSTEST_CLIG process after the process has been created, the test excludes it from further lock and file sharing tests. At the end of the cluster-integration test, an attempt is made to report any errors seen by that node.

UETCLIG00.EXE has two threads of execution: the primary and the secondary. The first, or primary thread, checks the cluster configuration (VMS nodes, HSC nodes, and the attached disks that are available to the node running the test). For selected VMS nodes, the primary thread attempts to start up a SYSTEST_CLIG process through DECnet–VAX. If the primary thread was able to start a SYSTEST_CLIG process on a node, the node runs the command file UETCLIG00.COM, which starts up UETCLIG00.EXE and runs the secondary execution thread.

The process running the primary thread checks to see that it can communicate with the processes running the secondary threads. It then instructs them to take out locks so that a deadlock situation is created.

The primary thread tries to create a file on some disk on selected VMS and HSC nodes in the cluster. The primary thread writes a block, reads it back, and verifies it. The primary thread selects one VMS node at random and asks that node to read the block and verify it. The primary extends the file by writing another block and has the secondary read and verify the second block. The file is deleted.

The secondary processes exit. They copy to the primary process the contents of their SYS$ERROR files, so that the UETP log file and console report show all problems in a central place. DECnet–VAX automatically creates a NETSERVER.LOG in SYS$TEST as the test is run, so that if necessary, you can read that file later from the node in question.

During the test run, the primary process uses the system service SYS$BRKTHRU to announce the beginning and ending of the test to each VMS node's console terminal.

You can define the group logical name MODE to the equivalence string DUMP to trace most events as they occur. Note that the logical name definitions apply only to the node on which they were defined. You must define MODE on each system in the VAXcluster on which you want to trace events.

11 Customizing VMS DECwindows Software

This chapter describes the tasks that you must perform to customize VMS DECwindows software. If you have not installed VMS DECwindows software on your system, skip this chapter. Depending on your configuration, VMS DECwindows customization tasks include the following:

- Controlling the startup of VMS DECwindows software as described in Section 11.1.
- Customizing the server startup as described in Section 11.2.
- Using VMS/ULTRIX Connection software as a DECwindows transport.
- Using customer-written transports with DECwindows.
- Defining print formats as described in Section 11.6.
- Compiling the optional Ada software as described in Section 11.7.

11.1 Controlling Startup of VMS DECwindows Software

If you installed VMS Version 5.4 on your system, go to Section 11.1.1.

If you upgraded your system to VMS Version 5.4, go to Section 11.1.2.

11.1.1 After an Installation

After you register the VMS license and reboot the system, DECwindows starts automatically (if you installed it). However, if you plan to use DECnet–VAX, you must edit SYS$SYSROOT:[SYSMGR]SYSTARTUP_V5.COM by adding a comment delimiter (!) immediately following the dollar-sign ($) in the following command:

```
$ DEFINE DECW$IGNORE_DECNET TRUE
```

This step was described in Section 4.5. If you have not already done so, perform this step now.

If you are not going to use DECnet–VAX, or have not yet registered the license, leave this command uncommented. This command tells VMS DECwindows not to wait for DECnet–VAX.

11.1.2 After an Upgrade

VMS DECwindows software starts automatically after SYSTARTUP_V5.COM runs, however you may need to do the following:

1 If you upgraded from V5.1, V5.1-1, or V5.1-B, you need to edit SYSTARTUP_V5.COM and delete the following line:

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

2 If you placed the DECW$STARTUP command in a file other than SYSTARTUP_V5.COM, you can either delete the command from the file it is in or add the following command to SYSTARTUP_V5.COM:

```
$ DEFINE DECW$IGNORE_DECWINDOWS TRUE ! Delay DECwindows startup
```

You should only add the above command to SYSTARTUP_V5.COM if your site-specific startup is going to execute DECW$STARTUP at a later time. You should execute DECW$STARTUP on all VAX computers using any function of DECwindows software, including those computers performing only as DECwindows servers. You do not need to execute DECW$STARTUP if you have removed *all* classes of DECwindows software using DECW$TAILOR.

3 If you plan to use the DECwindows example C programs, make sure that VAX C is installed using the option to extract VAX C definition (.h) files. All of the DECwindows example C programs assume that the definition files were extracted; if you have already installed VAX C and you did not extract the definition files, the DECwindows example C programs do not work. For more information, see the *VMS Version 5.4 Release Notes*.

11.1.3 VMS DECwindows Logical Names Created At Startup

After you start VMS DECwindows, a system-wide logical name table is created and is added to the default logical name table search lists. Table 11–1 lists the logical names contained in this table.

Table 11–1 DECW$LOGICAL_NAMES

Logical Name	Definition
CDA$LIBRARY	DDIF document style guide
DECW$BOOK	Online documentation
DECW$EXAMPLES	DECwindows example files directory
DECW$INCLUDE	DECwindows MIT-compatible C language include files
DECW$KEYMAP	DECwindows keyboard map definitions directory used by the Session Manager
DECW$SYSTEM_DEFAULTS	The system-wide directory where DECwindows application will look for resource files
DECW$USER_DEFAULTS	The user directory where DECwindows applications will look for resource files
DECW$WINMGREXE	Window manager executable file used by the Session Manager

(continued on next page)

Table 11–1 (Cont.) DECW$LOGICAL_NAMES

Logical Name	Definition
DECW$XLIBERRDB	Pointer to the Xlib error resource database
UIL$INCLUDE	Directory containing toolkit UIL files
VUE$LIBRARY	Directory containing FileView command files and the FileView default system profile
XDPS$INCLUDE	Directory containing Display Postscript programming include header files
XDPS$EXAMPLES	Directory containing Display Postscript example application files

11.2 Customizing the Server Startup

If you are not planning on starting the VMS DECwindows server, go to Section 11.6.

The startup command procedures automatically determine most of the configuration variables for the VMS DECwindows server. However, there are four items which the command procedures cannot determine:

- The startup procedures assume that you have a 75 dot per inch (dpi) monitor (such as a VR260 or VR290).
- If you have a VAXstation 2000/GPX, a VAXstation 3xxx/GPX, or a VAXstation II/GPX, the startup procedures assume that you have a color monitor.
- The startup procedures assume that you have a North American keyboard layout (you have an LK201-AA or -LA keyboard).
- The startup procedures assume that the server will use only DECnet–VAX and local transports.

If any of these assumptions is incorrect, you must override them as follows:

1. Make a copy of the file SYS$MANAGER:DECW$PRIVATE_SERVER_SETUP.TEMPLATE and rename it to a .COM file type. The comments in this file show how to customize individual workstations by adding a section of DCL commands for each workstation. You can also add DCL commands to the common section of the server startup file. By doing this, you ensure that the customizations you make affect all workstations that use the server startup file.
2. If your screen supports 100 dpi, add the following line:

   ```
   $ DECW$SERVER_DENSITY == 100
   ```

   Add this line to the section for each workstation that has a 100 dpi monitor, or add it to the common section for all workstations (if all the workstations have 100 dpi monitors).
3. The section for any monochrome /GPX workstation should contain the following line if you want to override the default:

   ```
   $ DECW$COLOR == "FALSE"
   ```

4 To override the default keyboard layout, determine the correct name from the model number of your LK201 keyboard, as follows:

a. Turn the keyboard upside down, and look for a label which specifies the model number. The model number should be in a format similar to LK201-xx.

The model number may also be listed as simply LK201. In this case, the information you need is in another part of the label where there will be a number that has the following format:

```
nn-nnnnn-xx.
```

b. Use the xx part of this number to choose a keymap name from Table 11–2. The table is arranged based on the dialect that each keyboard is designed for. Choose the keymap ending in _DP for the data processing keyboard layout, or the keymap ending in _TW for the typewriter layout.

Table 11–2 DECwindows Keymap

Dialect	Model	DECwindows Keymap Name
Austrian/German	LK201-(AG,LG,BG,MG)	AUSTRIAN_GERMAN_LK201LG_DP
		AUSTRIAN_GERMAN_LK201LG_TW
	LK201-(NG,PG)	AUSTRIAN_GERMAN_LK201NG_DP
		AUSTRIAN_GERMAN_LK201NG_TW
Belgian/French	LK201-(AP,LP,BP,MP)	BELGIAN_FRENCH_LK201LP_DP
		BELGIAN_FRENCH_LK201LP_TW
British	LK201-(AE,LE,BE,ME)	BRITISH_LK201LE_DP
		BRITISH_LK201LE_TW
Canadian/French	LK201-(AC,LC,BC,MC)	CANADIAN_FRENCH_LK201LC_DP
		CANADIAN_FRENCH_LK201LC_TW
Danish	LK201-(AD,LD,BD,MD)	DANISH_LK201LD_DP
		DANISH_LK201LD_TW
	LK201-(ED,RD,FD)	DANISH_LK201RD_DP
		DANISH_LK201RD_TW
Dutch	LK201-(AH,LH,BH,MH)	DUTCH_LK201LH_DP
		DUTCH_LK201LH_TW
	LK201-(NH,PH)	DUTCH_LK201NH
Finnish	LK201-(AF,LF,BF,MF)	FINNISH_LK201LF_DP
		FINNISH_LK201LF_TW
	LK201-(NF,PF)	FINNISH_LK201NF_DP
		FINNISH_LK201NF_TW
Flemish	LK201-(AB,LB,BB,MB)	FLEMISH_LK201LB_DP
		FLEMISH_LK201LB_TW

(continued on next page)

Table 11–2 (Cont.) DECwindows Keymap

Dialect	Model	DECwindows Keymap Name
Icelandic	LK201-(AU,LU,BU,MU)	ICELANDIC_LK201LU_DP
		ICELANDIC_LK201LU_TW
Italian	LK201-(AI,LI,BI,MI)	ITALIAN_LK201LI_DP
		ITALIAN_LK201LI_TW
North American	LK201-(AA,LA,BA,MA)	NORTH_AMERICAN_LK201LA
Norwegian	LK201-(AN,LN,BN,MN)	NORWEGIAN_LK201LN_DP
		NORWEGIAN_LK201LN_TW
	LK201-(EN,RN,FN)	NORWEGIAN_LK201RN_DP
		NORWEGIAN_LK201RN_TW
Portuguese	LK201-(AV,LV,BV,MV)	PORTUGUESE_LK201LV
Spanish	LK201-(AS,LS,BS,MS)	SPANISH_LK201LS_DP
		SPANISH_LK201LS_TW
Swedish	LK201-(AM,LM,BM,MM)	SWEDISH_LK201LM_DP
		SWEDISH_LK201LM_TW
	LK201-(NM,PM)	SWEDISH_LK201NM_DP
		SWEDISH_LK201NM_TW
Swiss/French	LK201-(AK,LK,BK,MK)	SWISS_FRENCH_LK201LK_DP
		SWISS_FRENCH_LK201LK_TW
Swiss/German	LK201-(AL,LL,BL,ML)	SWISS_GERMAN_LK201LL_DP
		SWISS_GERMAN_LK201LL_TW
Combined US/UK	LK201-(EE,RE,PE)	UK_LK201RE
		US_LK201RE

c. After you choose a keymap name, modify the server startup file. For example, to change the keyboard layout to a Dutch typewriter layout, add the following line to the server startup file:

```
$ DECW$DEFAULT_KEYBOARD_MAP == "DUTCH_LK201LH_TW"
```

Add a line similar to this for each workstation that does not have a North American keyboard layout, or add it to the common section for all workstations (if all the workstations use the same keyboard layout).

5 To add support for the Transmission Control Protocol/Internet Protocol (TCP/IP) transport using the VMS/ULTRIX Connection product (UCX), add the following line:

```
$ DECW$SERVER_TRANSPORTS == "DECNET,LOCAL,TCPIP"
```

Add this line to the section for each workstation that will be using TCP/IP, or add it to the common section for all workstations.

To include support for other transports, add the last part of the transport image name to the list of transports. For example, suppose you want to add support for a customer-written transport that has the following image name:

SYS$SHARE:DECW_TRANSPORT_FOO

You would add the following line:

```
$ DECW$SERVER_TRANSPORTS == "DECNET,LOCAL,FOO"
```

6 Reboot the workstations for which you customized the server startup.

7 Customize security as described in the *VMS DECwindows User's Guide*.

After you start the server, a per-server logical name table is created on your system but is not added to the default logical name table search lists. Table 11–3 lists the logical names.

Note: These logical names may change in a future release.

Table 11–3 DECW$SERVER(n)_TABLE

Logical Name	Definition
DECW$COLOR	Determines whether the server will consider a VAXstation/GPX to be color or intensity
DECW$DEFAULT_FONT	Determines the font which is initially loaded in a graphics context
DECW$DEFAULT_KEYBOARD_MAP	Determines which keyboard map file the server uses
DECW$FONT	Directory where the server will look for fonts
DECW$KEYBOARD	Device name for system keyboard
DECW$KEYMAP	DECwindows keyboard map definitions directory used by the server
DECW$MONITOR_DENSITY	Determines what monitor density the server will use to calculate the monitor size
DECW$POINTER	Device name for system pointer
DECW$RGB_xxx	Definitions of color names used by the XStoreNamedColor and XAllocNamedColor requests
DECW$SERVER_DISABLE_CH	If defined to be true, server exception will cause the server to crash and produce a dump file, rather than attempt to continue

(continued on next page)

Table 11–3 (Cont.) DECW$SERVER(n)_TABLE

Logical Name	Definition
DECW$SERVER_SCREENS	Device name used by the server. The first two characters indicate the device-dependent shareable image that is loaded for device support (for example, GCA0:)
DECW$SERVER_TRANSPORTS	A list of transport types that the DECwindows server loads when it starts up
DECW$XSIZE_IN_PIXELS	Specifies the width of the workstation screen
DECW$YSIZE_IN_PIXELS	Specifies the height of the workstation screen

11.3 Using TCP/IP with DECwindows

DECwindows contains support for the TCP/IP transport. This support depends on the VMS/ULTRIX Connection software (UCX), which is a Digital layered product. Before you can use DECwindows TCP/IP transport interface, you must install UCX. For more information about TCP/IP concepts, see the *VMS/ULTRIX Connection System Manager's Guide*.

Note: Other vendor's implementation of TCP/IP transports use different call interfaces. For this reason, DECwindows support for TCP/IP is compatible only with UCX.

You can conserve memory and process slots by configuring UCX software for the minimum DECwindows requirement to support the X protocol. DECwindows only requires that INET_ACP be running. DECwindows does not require that the NFS server, the FTPD server, or the remote terminal servers be running. For information on how to configure the UCX software, see the *VMS/ULTRIX Connection Installation Guide*.

11.4 Using Customer-Written Transports with DECwindows

DECwindows supports DECnet–VAX, local, and TCP/IP transports. However, you can use other transports with DECwindows by writing a DECwindows transport interface layer. Digital will likely supply new transports for various layered products, and third-party vendors can also supply new transports, such as other TCP/IP transports.

For information on writing a transport interface, see the *VMS DECwindows Transport Manual*.

The transport interface image's filename must have the following format where *name* is the unique transport name:

DECW$TRANSPORT_name (for transports supplied by Digital)
DECW_TRANSPORT_name (for customer-written transports)

Install the image as a protected image in the SYS$SHARE directory (as defined by an executive mode logical name). Digital recommends that you install the image by adding the following line to SYS$MANAGER:SYSTARTUP_V5.COM:

```
$ INSTALL CREATE SYS$SHARE:DECW_TRANSPORT_name/OPEN/SHARED/HEADER_RES/PROTECTED
```

You can also include this command in a startup file that SYSTARTUP_V5 calls.

To use this transport with the DECwindows server, customize the server startup as described in Section 11.2.

11.5 Using the Example Transport

DECwindows has an example transport interface based upon the UCX software. It is similar to the DECwindows TCP/IP transport interface, except it includes its own source files and it uses a different port number (5000). You can run the example transport interface simultaneously with the TCP/IP transport interface.

If you have installed UCX software, you can install and use the example transport by doing the following:

1. Copy the example transport to the SYS$SHARE directory as follows:

```
$  COPY DECW$EXAMPLES:DECW$TRANSPORT_EXAMPLE.EXE SYS$SHARE:/PROT=W:RE
```

2. Add the following line to SYS$MANAGER:SYSTARTUP_V5.COM:

```
$ INSTALL CREATE SYS$SHARE:DECW$TRANSPORT_EXAMPLE /OPEN /SHARED/ HEADER_RES /PROTECTED
```

3. Customize the private server startup as described in Section 11.2 for TCP/IP, except use the transport name ‘‘EXAMPLE’’. For example, you could add the following line:

   ```
   $ DECW$SERVER_TRANSPORTS == "DECNET,LOCAL,TCPIP,EXAMPLE"
   ```

4. Reboot the workstation.

5. Customize security as described in the *VMS DECwindows User's Guide* for TCP/IP, but use the transport name ‘‘EXAMPLE’’ instead of ‘‘TCPIP’’.

6. Create a DECterm window on the workstation and enter the following command:

   ```
   $  SET DISPLAY /CREATE /NODE=your_node /TRANSPORT=EXAMPLE
   ```

   Substitute the node name of your workstation for *your_node*.

7. Run any DECwindows application from the DECterm window. It will now use the example transport to display graphics on your workstation screen. For example, you would enter the following command to run the clock:

   ```
   $  RUN SYS$SYSTEM:DECW$CLOCK
   ```

11.6 Defining Print Formats

If you do not want to define print formats on your system, go to Section 11.7. The print dialog box is used in many applications for the queuing of files or screens to a printer. (See the *VMS DECwindows User's Guide* for more information about the print dialog box.) By default, all printing devices on the system are displayed in the ''Printers'' listbox. However, you can take advantage of a feature where print queues can be associated with print formats through the definition of logical names. The logical name is defined to be a comma-separated or space-separated list of print queues with the first queue being the default choice.

For example, if DECW$PRINTER_FORMAT_ANSI2 is defined to be ''MY_ PRINTER, CLUSTER_LN03, ANOTHER_LN03'', when you select ''ANSI2'' from the print format listbox, only MY_PRINTER, CLUSTER_LN03, and ANOTHER_LN03 will be shown in the printer listbox, with MY_PRINTER being the default choice.

Table 11–4 lists the logical names and the print formats associated with each.

Table 11–4 Print Dialog Box Logical Names

Logical name	Print format
DECW$PRINTER_FORMAT_TEXT	Text
DECW$PRINTER_FORMAT_LINE	Line Printer
DECW$PRINTER_FORMAT_TERM	Terminal
DECW$PRINTER_FORMAT_ANSI2	ANSI2
DECW$PRINTER_FORMAT_ANSI	ANSI
DECW$PRINTER_FORMAT_REGIS	ReGIS
DECW$PRINTER_FORMAT_TEK	Tektronix
DECW$PRINTER_FORMAT_DDIF	DDIF

The logical names can be defined by system managers in the appropriate logical name tables and can be overridden by users.

If you have defined any of the logical names in Table 11–4, they will serve as the list of printer queue names to be presented when the corresponding print format is selected. This method is much faster than making the print dialog box derive the names of all the queues on the system, most of which are inappropriate for the print format under consideration, as the following example shows:

```
 $ DEFINE DECW$PRINTER_FORMAT_TEXT   "CLUSTER_LN03,CLUSTER_PRINT, -
_$ ANSI_ARTWRK,ANSI_PROTON"
$ DEFINE DECW$PRINTER_FORMAT_LINE   "CLUSTER_PRINT"
$ DEFINE DECW$PRINTER_FORMAT_ANSI2  "CLUSTER_LN03,ANSI_ARTWRK,ANSI_PROTON"
$ DEFINE DECW$PRINTER_FORMAT_ANSI   "CLUSTER_LN03,ANSI_ARTWRK,ANSI_PROTON"
$ DEFINE DECW$PRINTER_FORMAT_REGIS  "SYS$NULL"
```

11.7 Compiling Ada Source Files

If you have not installed Ada programming software on your system, skip this section. VMS DECwindows provides programming interface definitions for the Ada language. When you select Ada support at the time of the VMS DECwindows kit installation, four Ada package source files are placed in the SYS$LIBRARY: directory of your system. These files are:

- CDA$CDA_.ADA—Package CDA—Compound Document Architecture
- DDIF$DDIF_.ADA—Package DDIF—Digital Document Interchange Format
- DECW$DWT_.ADA—Package DWT—DECwindows Toolkit
- DECW$X_.ADA—Package X—Xlib

These package source files can be individually compiled into your Ada program libraries or compiled into the systemwide Ada predefined library. To make the packages available systemwide, the command file SYS$UPDATE:DECW$COMPILE_ADA_UNITS.COM is provided.

This command procedure compiles all four packages into the predefined Ada library, and, if the VAX Source Code Analyzer (SCA) product is present, loads SCA analysis data for the packages into the SCA library for the predefined library. The command procedure should be run as a batch job and should have available a minimum of 2000 pages in the working set; however, 3000 pages is preferable. A page file quota of at least 30000 pages is suggested.

Once the units are compiled into the predefined Ada library, you must execute the following Ada program library manager command to make the units visible:

```
$ ACS ENTER UNIT ADA$PREDEFINED CDA,DDIF,DWT,X
```

You need to do this only once. This step is also performed automatically for all Ada program libraries created after the VMS DECwindows units are compiled into the predefined library.

Future installations of VMS DECwindows might replace the Ada packages. If so, the new packages must be compiled as shown. If you have already entered the units into your own library, you must then execute the following command to make your library current:

```
$ ACS REENTER *
```

Future installations of VAX Ada might replace the Ada predefined library and remove the VMS DECwindows units. If this occurs, execute the DECW$COMPILE_ADA_UNITS.COM command procedure again.

12 Using VMSTAILOR and DECW$TAILOR

Read this chapter if you want to add or remove optional VMS software or VMS DECwindows software from your system disk. To add or remove optional VMS files, use VMSTAILOR. To add or remove DECwindows software, use DECW$TAILOR.

12.1 Using VMSTAILOR

To add or remove unwanted VMS files from the system disk, use VMSTAILOR. For example, if you are not running DECnet–VAX, and you do not need the network support files, you can remove those files using VMSTAILOR. If you later decide you want to run DECnet–VAX, you can add the necessary files at any time using VMSTAILOR and your distribution media.

VMSTAILOR has three phases:

1. VMSTAILOR asks if you want to tailor your system by adding files (tailor on) or by deleting files (tailor off).
2. If you choose to tailor files on, VMSTAILOR lists each class of VMS files and asks if you want to include them on the system disk. If you choose to tailor files off, VMSTAILOR lists each class of files and asks if you want to delete them from the system disk.
3. VMSTAILOR adds or deletes the files you indicated from the system disk. If you are tailoring files on, you must load your VMS distribution media in a drive available to the system.

The following table lists the classes and subclasses of VMS files that can be added or deleted using VMSTAILOR. For a complete list of VMS files and their functions, see the *Guide to Setting Up a VMS System.*

Table 12–1 Classes of VMS Files That Can Be Added or Removed with VMSTAILOR

Network Support
Incoming Remote File Access files
Incoming Remote Terminal files
Network Test files
Remote Task Loading

(continued on next page)

Table 12–1 (Cont.) Classes of VMS Files That Can Be Added or Removed with VMSTAILOR

Programming Support
Debugger utility (DEBUG) Image Dump utility (ANALYZE/PROCESS_DUMP) RMS Analyze and FDL Editor utilities (ANALYZE/RMS, ANALYZE/FDL) Message utility (MESSAGE) Object and Shareable Image libraries Macro libraries Macro assembler SDL intermediary form of STARLET.MLB FORTRAN require files VAX-C object libraries
RMS Journaling Files
No subclasses in this tailor class
System Programming Support
Files–11 ODS1 ACP Monitor utility Analyze Object File utility (ANALYZE/IMAGE, ANALYZE/OBJECT) Delta debugger System Dump Analyzer utility (ANALYZE/SYSTEM, ANALYZE/CRASH) System Symbol Table file (SYS.STB) Miscellaneous Symbol Table files
Secure User's Environment
File Access Control List utilities Print and Batch Queue utilities Input Queue Symbiont (Card Reader) Accounting Log Report Generator utility DECdtm

(continued on next page)

Table 12–1 (Cont.) Classes of VMS Files That Can Be Added or Removed with VMSTAILOR

Utilities
MAIL utility DUMP utility RUNOFF utility PHONE utility VMS HELP library VMS System Messages Help Library Foreign Terminal Support LAT-11 terminal server (via Ethernet) Standalone BACKUP Error Log Report Generator utility (ANALYZE/ERROR) VAXTPU utility Terminal Fallback Facility TECO editor EDT documentation National Character Set utility (NCS)
VMS Workstation Support
Workstation device support
BLISS Require Files
No subclasses in this tailor class
Miscellaneous Files
System map LPA-11 support
User Environment Test Package
No subclasses in this tailor class
Example Files
No subclasses in this tailor class

12.1.1 Removing VMS Files

To remove unwanted files using VMSTAILOR, perform the following steps:

1. Log in to the SYSTEM account.
2. Enter the following command, and press RETURN:

```
$ RUN SYS$UPDATE:VMSTAILOR
```

VMSTAILOR displays a description of the VMSTAILOR program and displays the following message:

```
Do you want to tailor files "ON" or "OFF"?
```

3 Type OFF to remove unwanted files, and press RETURN.

The VMSTAILOR program lists each group, or *class*, of files and its size in blocks. Files are grouped according to their function. For example, all the files required for network support are in one class. A file class is made up of many smaller groups called *subclasses*. You can eliminate an entire class of files, or you can eliminate one or more of its subclasses. For example:

```
CLASS - Network support
          Size of entire class (with subclasses): 1499
          Size of common files required for any subclass: 1249
          Do you wish to select the entire class (default = NO)?
   .
   .
   .
```

4 Decide which file classes or subclasses you do not need to support your system. The VMSTAILOR program displays step-by-step instructions. Follow these instructions to specify which classes or subclasses of files you want to remove.

5 VMSTAILOR deletes the files you selected and displays the names of those files. After it finishes, AUTOGEN runs automatically to make the adjustments that are necessary after system files are deleted. AUTOGEN also reboots the system.

Caution: To cancel VMSTAILOR during the first and second phases, press CTRL/C, CTRL/Y, or CTRL/Z. If you cancel VMSTAILOR during the third phase, you may end up with a partially tailored disk. For a description of each phase, see Section 12.1.

12.1.2 Adding VMS Files

To add optional VMS files that you previously removed, or that you chose not to install during the VMS installation procedure, do the following:

1 Log in to the SYSTEM account.

2 Load your distribution media in a drive available to the system. For example, if your distribution media is on a TK50 tape cartridge, load the tape cartridge labeled *VMS V5.4 BIN TK50 1/2 VMS BINARY* in a TK50 drive available to the system. Make sure the drive is online and ready.

3 Type the following command, and press RETURN:

```
$ RUN SYS$UPDATE:VMSTAILOR
```

VMSTAILOR displays a description of the VMSTAILOR program and displays the following message:

```
Do you want to tailor files "ON" or "OFF"?
```

4 Type ON to add files, and press RETURN.

The VMSTAILOR program lists each group, or *class*, of files and its size in blocks. Files are grouped according to their function. For example, all the files required for network support are in one class. A file class is made up of many smaller groups called *subclasses*. You can add an entire class of files, or you can add one or more of its subclasses. For example:

```
CLASS - Network support
          Size of entire class (with subclasses): 1499
          Size of common files required for any subclass: 1249
          Do you wish to select the entire class (default = NO)?
   .
   .
   .
```

5 Decide which file classes or subclasses you need to add. VMSTAILOR displays step-by-step instructions. Follow these instructions to specify which classes or subclasses of files you want to add.

6 VMSTAILOR adds the files you specified and displays the names of the files. After the files have been added, AUTOGEN runs automatically to make the adjustments that are necessary after system files are deleted. AUTOGEN also reboots the system.

Caution: To cancel VMSTAILOR during the first and second phases, press CTRL/C, CTRL/Y, or CTRL/Z. If you cancel VMSTAILOR during the third phase, you may end up with a partially tailored disk. For a description of each phase, see Section 12.1.

After adding files to the system disk, you should apply any updates that affect them.

For example, suppose you do not need the VMS Version 5.4 PHONE utility and you run VMSTAILOR to remove those files. Later on, if you decide you want to use PHONE, you can run VMSTAILOR to return the PHONE files to the system disk. You then apply any VMS update that has occurred since Version 5.4 that affected the PHONE utility. VMS update procedures create a .TXT file that contains a description of patched files. This file usually has the following format: SYS$UPDATE:VMSUn05n.TXT.

Example 12–1 illustrates how to add all Network Support files, and selected Programming Support files to a system disk.

Using VMSTAILOR and DECW$TAILOR

12.1 Using VMSTAILOR

Example 12–1 Sample VMSTAILOR Session

```
$  RUN SYS$UPDATE:VMSTAILOR
   .
   .
   .
 Do you want to tailor files "ON" or "OFF"?  ON

TAILOR-ON
---------
 You will now be prompted with a list of the classes and subclasses
 of VMS files that are optional. The size of each class and subclass is included
 in the list. This will help you decide whether or not you want to add
 a class or subclass to your system.

 Under some classes, there is a set of common files that is required in order
 for any subclasses to work. These files are added when you ask
 for either the entire class or any of its subclasses.

Total size of the system disk is 2376153 blocks.
Total space used on the system disk is 2241252 blocks.
Total space left on the system disk is 134901 blocks.


CLASS - Network support
          Size of entire class (with subclasses): 1499
          Size of common files required for any subclass: 1249
          Do you wish to select the entire class (default = NO)?  YES

CLASS - Programming Support
          Size of entire class (with subclasses): 14706
          Size of common files required for any subclass: 0
          Do you wish to select the entire class (default = NO)?  Return
          Do you wish to select any of its subclasses (default = NO)? ) YES
          Select the subclasses that you wish to provide:
             SUBCLASS - Debugger utility (DEBUG) (4690 blocks) [NO]:  YES
             SUBCLASS - Image Dump utility (ANALYZE/PROCESS_DUMP) (48 blocks) [NO]:   YES
             SUBCLASS - RMS Analyze and FDL Editor utilities (ANALYZE/RMS, ANALYZE/FDL) (374 blocks) [NO]:  YES
             SUBCLASS - Message utility (MESSAGE) (49 blocks) [NO]:  Return
             SUBCLASS - Object and Shareable Image libraries (4311 blocks) [NO]:  YES
             SUBCLASS - Macro libraries (2293 blocks) [NO]:  Return
             SUBCLASS - Macro assembler (457 blocks) [NO]:  Return
             SUBCLASS - SDL intermediary form of STARLET.MLB (1957 blocks) [NO]:  Return
             SUBCLASS - FORTRAN require files (51 blocks) [NO]:  Return
             SUBCLASS - VAX-C object libraries (471 blocks) [NO]:  Return

CLASS - RMS journaling files
          Size of entire class (with subclasses): 155
          No subclasses in this tailor class.
          Do you wish to select the entire class (default = NO)?  Return

CLASS - System programming Support
          Size of entire class (with subclasses): 2286
          Size of common files required for any subclass: 93
          Do you wish to select the entire class (default = NO)?  Return
          Do you wish to select any of its subclasses (default = NO)?  Return

CLASS - Secure User's Environment
          Size of entire class (with subclasses): 919
          Size of common files required for any subclass: 0
          Do you wish to select the entire class (default = NO)?  Return

          Do you wish to select any of its subclasses (default = NO)?  Return

   .
   .
   .

Files have been selected
Do you wish to add all of the options selected?  YES
Creating temporary command file, please wait...
* Enter device containing VMS distribution media: mua0:

   .
   .
   .
```

12.2 Using DECW$TAILOR

Use DECW$TAILOR to add or remove some or all DECwindows files from the system disk. For example, if you are not planning on using the DECwindows programming support, you can remove those files. Later, if you decide you want DECwindows programming support, you can add the files to the system disk using DECW$TAILOR and your VMS DECwindows distribution media.

DECW$TAILOR has three phases:

1. DECW$TAILOR asks if you want to tailor your system by adding files (tailor on) or by deleting files (tailor off).
2. If you choose to tailor files on, DECW$TAILOR lists each class of DECwindows files and asks if you want to include them on the system disk. If you choose to tailor files off, DECW$TAILOR lists each class of DECwindows files and asks if you want to delete them from the system disk.
3. DECW$TAILOR adds or deletes the files you indicated from the system disk. If you are tailoring files on, you must load your VMS DECwindows distribution media in a drive available to the system. For example, if your VMS DECwindows distribution kit is on a TK50 tape cartridge, load the tape cartridge labeled "VMS V5.4 BIN TK50 2/2 S/A BKUP - DECWINDOWS" into a TK50 drive available to the system.

The following table lists the classes and subclasses of VMS DECwindows files that you can add or delete using DECW$TAILOR. For a list of DECwindows files and their functions, see the current version of the *VMS Upgrade and Installation Manual*.

Table 12–2 Classes of VMS Files That Can Be Added or Removed with DECW$TAILOR

DECwindows Applications Files
No subclasses in this tailor class
DECwindows Device Support Files[1]
PEX for PHIGS support files
75 Dots Per Inch Video Font Files
No subclasses in this tailor class

[1] If you tailor device support files ON or OFF, the system reboots

(continued on next page)

Table 12–2 (Cont.) Classes of VMS Files That Can Be Added or Removed with DECW$TAILOR

100 Dots Per Inch Video Font Files
No subclasses in this tailor class
DECwindows Programming Support Files
Ada files BASIC files BLISS files VAXC (MIT binding) files VAXC (VAX binding) files FORTRAN files Macro files Pascal files PL/1 files
Example Files
No subclasses in this tailor class

12.2.1 Removing DECwindows Files

To remove DECwindows files from the system disk, do the following:

1. Log in to the SYSTEM account.
2. Type the following command, and press RETURN:

   ```
   $  RUN SYS$UPDATE:DECW$TAILOR
   ```

   DECW$TAILOR displays information about the DECW$TAILOR program and displays the following message:

   ```
   Do you want to tailor files "ON" or "OFF"?
   ```

3. Type OFF to remove unwanted files, and press RETURN.

   The DECW$TAILOR program lists each group, or *class*, of files and its size in blocks. Files are grouped according to their function. For example, all the files required for VMS DECwindows device support are in one class. A file class is made up of many small subclasses. You can eliminate an entire class of files, or you can eliminate one or more of its subclasses. For example:

   ```
   CLASS - DECwindows device support files
             Size of entire class (with subclasses): 7868
             Size of common files required for any subclass: 5847
             Do you wish to select the entire class (default = NO)?
      .
      .
      .
   ```

4 Decide which file classes or subclasses you do not need to support your system. DECW$TAILOR displays step-by-step instructions. Follow these instructions to specify which of the classes or subclasses you want to remove.

5 DECW$TAILOR deletes the files you selected and displays the names of those files. If you deleted device support files, the system automatically shuts down and reboots.

Caution: To cancel DECW$TAILOR during the first and second phases, press CTRL/C, CTRL/Y, or CTRL/Z. If you cancel DECW$TAILOR during the third phase, you may end up with a partially tailored disk. For a description of each phase, see Section 12.2.

12.2.2 Adding DECwindows Files

To install DECwindows after the VMS installation, or add optional DECwindows files that you previously chose not to install or that you removed, do the following:

1 Log in to the SYSTEM account.

2 Load the distribution media containing the VMS DECwindows save sets in a drive available to your system. For example, if your distribution media is a TK50 tape cartridge, load the tape cartridge labeled *VMS V5.4 BIN TK50 2/2 DECwindows & S/A BKUP* in a TK50 drive available to the system. Make sure the drive is online and ready.

3 Enter the following command, and press RETURN:

```
$ RUN SYS$UPDATE:DECW$TAILOR
```

DECW$TAILOR displays information about the DECW$TAILOR program, and displays the following message:

```
Do you want to tailor files "ON" or "OFF"?
```

4 Type ON to add files, and press RETURN.

The DECW$TAILOR program lists each group, or *class*, of files and its size in blocks. Files are grouped according to their function. For example, all the files required for VMS DECwindows device support are in one class. A file class is made up of many small subclasses. You can add an entire class of files, or you can add one or more of its subclasses. For example:

```
CLASS - DECwindows device support files
          Size of entire class (with subclasses): 7868
          Size of common files required for any subclasses: 5847
          Do you wish to select the entire class (default = NO)?
   .
   .
   .
```

5 Decide which file classes or subclasses you do not need to support your system. The DECW$TAILOR program displays step-by-step instructions. Follow these instructions to specify which of the classes or subclasses you want to add.

6 DECW$TAILOR installs the files you selected and displays the names of those files. If you added device support files, the system automatically shuts down and reboots.

Caution: To cancel DECW$TAILOR during the first and second phases, press CTRL/C, CTRL/Y, or CTRL/Z. If you cancel DECW$TAILOR during the third phase, you may end up with a partially tailored disk. For a description of each phase, see Section 12.2.

After adding files to the system disk, apply any updates that affect them. For example, suppose you delete the Version 5.4 DECwindows 100 dot-per-inch video font files. Later on, you are able to use the 100 dot-per-inch fonts, so you run DECW$TAILOR to return the files to the system disk. You must then apply any VMS update that has occurred since Version 5.4 that affected the 100 dot-per-inch video font files. VMS update procedures create a .TXT file that contains a description of modified files. This file is usually named with the following format: SYS$UPDATE:VMSUn05n.TXT.

Example 12–2 illustrates how to add the files for DECwindows applications, and the DECwindows programming support files for Ada and BLISS to a system disk.

Example 12–2 Sample DECW$TAILOR Session

```
$  RUN SYS$UPDATE:DECW$TAILOR
   .
   .
   .

 Do you want to tailor files "ON" or "OFF"?  ON

TAILOR-ON
---------
 You will now be prompted with a list of the classes and subclasses
 of DECwindows files that are optional. The size of each class and subclass is
 included in the list. This will help you decide whether or not you want to add
 a class or subclass to your system.

 Under some classes, there is a set of common files that is required in order
 for any subclasses to work. These files are added when you ask
 for either the entire class or any of its subclasses.

Total size of the system disk is 2376153 blocks.
Total space used on the system disk is 1728072 blocks.
Total space left on the system disk is 648081 blocks.


CLASS - DECwindows applications files
          Size of entire class (with subclasses): 11545
          No subclasses in this tailor class.
          Do you wish to select the entire class (default = NO)?  YES


CLASS - DECwindows device support files
          Size of entire class (with subclasses): 7868
          Size of common files required for any subclass: 5847
          Do you wish to select the entire class (default = NO)?  NO
          Do you wish to select any of its subclasses (default = NO)?  Return

CLASS - 75 dots per inch video font files
          Size of entire class (with subclasses): 2249

          No subclasses in this tailor class.
          Do you wish to select the entire class (default = NO)?  Return
```

(continued on next page)

Example 12–2 (Cont.) Sample DECW$TAILOR Session

```
CLASS - 100 dots per inch video font files
          Size of entire class (with subclasses): 2788
          No subclasses in this tailor class.
          Do you wish to select the entire class (default = NO)?  Return

CLASS - DECwindows programming support files
          Size of entire class (with subclasses): 26206
          Size of common files required for any subclass: 924
          Do you wish to select the entire class (default = NO)?  Return
          Do you wish to select any of its subclasses (default = NO)?  YES
          Select the subclasses that you wish to use:
             SUBCLASS - Ada files (3365 blocks) [NO]:  YES
             SUBCLASS - BASIC files (2694 blocks) [NO]:  Return
             SUBCLASS - BLISS files (2083 blocks) [NO]:  YES
             SUBCLASS - VAX C (MIT binding) files (1302 blocks) [NO]:  Return
             SUBCLASS - VAX C (VAX binding) files (2945 blocks) [NO]:  Return
             SUBCLASS - FORTRAN files (2034 blocks) [NO]:  Return
             SUBCLASS - Macro files (3641 blocks) [NO]:  Return
             SUBCLASS - Pascal files (4662 blocks) [NO]:  Return
             SUBCLASS - PL/1 files (2556 blocks) [NO]:  Return
Files have been selected
Do you wish to add all of the options selected?  YES
Creating temporary command file, please wait...
* Enter device containing DECwindows distribution media: MUA0:
   .
   .
   .
```

A Booting from [SYSF] During an Upgrade

This appendix describes how to boot the various VAX computers from [SYSF]. You must boot from the [SYSF] root during phase 2, 3, and 4 of the upgrade procedure. If you need more information on booting, refer to the upgrade and installation supplement for your VAX computer.

A.1 MicroVAX and VAXstation Computers

The information in this section applies to the following VAX computers:

VAXstation II, VAXstation II/GPX, and MicroVAX II
VAXstation 2000 and MicroVAX 2000
VAXstation 3100 and MicroVAX 3100 series
VAXstation 3200 and 3500
VAXstation 3520 and 3540
MicroVAX 3300, 3400, 3500, 3600, 3800, and 3900

To boot from [SYSF], enter the BOOT command in the following format and press the Return key:

```
>>> B/F0000000 device-name
```

Substitute the device name of the system disk for *device-name*.

A.2 VAX 8530, 8550, 8810 (8700), 8820-N (8800), 8820, 8830, and 8840

To boot from [SYSF], use the following procedure:

1. Press Ctrl/P. On the VAX 8530, VAX 8550, VAX 8810 (8700), and VAX 8820-N (8800), enter the following command at the console-mode prompt (>>>) and press the Return key:

   ```
   >>> HALT
   ```

   On the VAX 8820, VAX 8830, and VAX 8840, enter the following command at the PS-OS-0> prompt and press Return:

   ```
   PS-OS-0> HALT/CPU=ALL
   ```

2. On VAX 8530, VAX 8550, VAX 8810 (8700), and VAX 8820-N (8800), enter the BOOT command at the console-mode prompt (>>>) in the following format:

   ```
   >>> B dddn /R5:F0000000
   ```

   Substitute BCI, BDA, or UDA for *ddd*. Substitute the unit number of the drive holding the system disk for *n*.

   On the VAX 8820, 8830, and 8840 enter the BOOT command at the PS-CIO-0> prompt in the following format:

   ```
   PS-CIO-0> B dddn /R5=F0000000
   ```

Substitute BCI or BDA for *ddd*. Substitute the unit number of the drive holding the system disk for *n*.

A.3 VAX 11/750, VAX 8200, 8250, 8300, and 8350

To boot from [SYSF], create a command procedure named SYFBOO.CMD.

To create SYFBOO.CMD, use the following procedure:

Note: This procedure assumes that you have installed and booted the VMS operating system and are logged into the SYSTEM account.

1. Make sure that the console media is in the console drive.
2. To connect the console drive to the system, enter the following commands and press the Return key after each one:

```
$  RUN SYS$SYSTEM:SYSGEN
SYSGEN>  CONNECT CONSOLE
SYSGEN>  EXIT
```

3. Use the Exchange Utility to copy the appropriate boot command procedure from the console media to your current directory on the system disk. If the system disk is on an HSC drive and you have created DEFBOO.CMD, enter the following command and press Return:

```
$  EXCHANGE COPY CSA1:DEFBOO.CMD *
```

   On the VAX 11/750, if the system disk is on a local drive, copy the boot command procedure that matches the drive that holds the system disk. For example, suppose the system disk is on an RP06 drive and has a controller designation of A and a unit number of one. Enter the following command and press the Return key:

```
$  EXCHANGE COPY CSA1:DA1BOO.CMD *
```

4. Edit the boot command procedure. Change the line that deposits a value in register 5 (R5). This line contains the comment *!software boot flags*. The value is a hexadecimal number with eight digits. For example:

```
D/G R5 0    !software boot flags
```

   Change the left-most digit of the value to reflect the name of the root directory from which you want to boot. For example, to boot from the [SYSF] root directory, change the line as follows:

```
D/G/L R5 F0000000     !designated root is SYSF
```

5. Rename the boot command procedure to SYFBOO.CMD. For example, if the boot command procedure is DEFBOO.CMD, enter the following command and press the Return key:

```
$  RENAME DEFBOO.CMD SYFBOO.CMD
```

6. Use the Exchange Utility to copy SYFBOO.CMD back to the console media. Enter the following command and press Return:

```
$  EXCHANGE COPY SYFBOO.CMD CSA1:SYFBOO.CMD
```

7 When you are finished, enter the following command and press Return:

```
$ DISMOUNT CSA1
```

8 To secure the console media from unauthorized access, you must enter the following command and press Return:

```
$ MOUNT/FOREIGN/SYSTEM/NOWRITE/NOASSIST CSA1
```

To boot from [SYSF], use the following procedure:

1 Make sure the console media is in the console drive. On the VAX 11/750, make sure the keylock switch is set to LOCAL. Set the BOOT DEVICE switch to position A.

On the VAX 8200, set the upper keylock switch to ENABLE and the lower keylock switch to HALT.

2 Press Ctrl/P.

3 On the VAX 11/750, enter the following command and press the Return key:

```
>>> B/800 DDA0
```

On the VAX 8200, enter the following command at the console-mode prompt (>>>) and press Return:

```
>>> B/R5:800 CSA1
```

4 At the BOOT58> prompt, enter the following command:

```
BOOT58> @SYFBOO.CMD
```

A.4 VAX 11/730, VAX 11/780, VAX 11/785, VAX 8600, and VAX 8650

To boot from [SYSF], create a command procedure named SYFBOO.CMD (for VAX 11/730, VAX 11/780, and VAX 11/785) or SYFBOO.COM (for VAX 8600 and VAX 8650).

To create SYFBOO.CMD or SYFBOO.COM, use the following procedure:

1 Make sure that the console media is in the console drive. On the VAX 8600, make sure the TERMINAL CONTROL switch is set to LOCAL.

2 Log into the SYSTEM account.

3 To connect the console drive, enter the following commands and press the Return key after each one:

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> CONNECT CONSOLE
SYSGEN> EXIT
```

4 Use the Exchange Utility to copy DEFBOO.CMD (for VAX 11/730, VAX 11/780, and VAX 11/785) or DEFBOO.COM (for VAX 8600 and VAX 8650) from the console media to your current directory on the system disk. On the VAX 11/780, VAX 11/785, and VAX 8600, enter a command similar to the following and press Return:

```
$ EXCHANGE COPY CSA1:DEFBOO.CMD *
```

On the VAX 11/730, enter the following command and press the Return key:

```
$  EXCHANGE COPY CSA2:DEFBOO.CMD *
```

5 Edit DEFBOO.CMD or DEFBOO.COM. Change the line that deposits a value in register 5 (R5). On the VAX 11/730, VAX 11/780, and 11/785, this line contains the comment *!software boot flags*. On the VAX 8600, this line contains the comment *!Use R5 for optional boot control flags [SYSB.]*. The value is a hexadecimal number with eight bits. For example:

```
DEPOSIT R5 10000000     !software boot flags
```

Change the left-most digit of the value to reflect the name of the root directory from which you want to boot. For example, to boot from [SYSF], change the line as follows:

```
DEPOSIT R5 F0000000     !software boot flags
```

6 Rename DEFBOO.CMD to SYFBOO.CMD (or DEFBOO.COM to SYFBOO.COM), as follows:

```
$  RENAME DEFBOO.CMD SYFBOO.CMD
```

7 Use the Exchange Utility to copy SYFBOO.CMD or SYFBOO.COM to the console media. Enter a command similar to the following:

```
$  EXCHANGE COPY SYFBOO.CMD CSA1:SYFBOO.CMD
```

For example, on the VAX 11/730 enter the following command and press the Return key:

```
$  EXCHANGE COPY SYFBOO.CMD CSA2:SYFBOO.CMD
```

8 When you are finished, enter a command similar to the following and press Return:

```
$  DISMOUNT CSA1
```

9 To secure the console media from unauthorized access, you must enter a command similar to the following and press Return:

```
$  MOUNT/FOREIGN/SYSTEM/NOWRITE/NOASSIST CSA1
```

To boot from [SYSF] during an upgrade, use the following procedure:

1 Make sure the console media is in the console drive. On the VAX 11/730, VAX 11/780, and VAX 11/785, make sure the keylock switch is set to LOCAL. On the VAX 8600, make sure the TERMINAL CONTROL switch is set to LOCAL.

2 Press Ctrl/P. On the VAX 11/780, VAX 11/785 and VAX 8600, enter the HALT command at the console-mode prompt (>>>) and press the Return key:

```
>>>  HALT
```

3 Use SYFBOO.CMD to boot from [SYSF]. Enter the following command and press Return:

```
>>>  B SYF
```

A.5 VAX 6000 Series

To boot from [SYSF], use the following procedure:

1. If you have a CIBCA-A adapter and are booting over the CI, insert the console tape cartridge in the console drive.
2. Press Ctrl/P to put the system in console mode.
3. Enter the BOOT command in the following format:

```
>>> BOOT /R5:F0000000  /XMI:a  /BI:b  [/R3:c]  [/NODE:d]  DUu
```

   where:

   - *a* is the XMI node number of the system disk.
   - *b* is the VAXBI node number of the system disk.
   - *c* pertains to Volume Shadowing. This qualifier is not required unless you are using Volume Shadowing. For more information, see the *VAX Volume Shadowing Manual*.
   - *d* is the HSC node number of the node being accessed. The /NODE qualifier is not necessary if you are booting from a local disk. The HSC node number is in hexadecimal. You can deposit a maximum of two HSC node numbers (if two HSCs are available).
   - *u* is the unit number of the drive holding the system disk.

   For example, suppose you want to boot from [SYSF] on an HSC disk with a unit number of one and your system has the following configuration:

   - The disk is connected to the VAXBI at node 2.
   - The VAXBI you are using is connected to the XMI at node 3.
   - The disk is available to two HSCs, node numbers 0E and 02.

   Enter the following command and press the Return key:

```
>>> BOOT/R5:F0000000/XMI:3/BI:2/NODE:0E02 DU1
```

A.6 VAX 9000 Series

The upgrade procedure automatically updates DEFBOO.CMD to boot from [SYSF]. If you want to manually boot a VAX 9000 computer from [SYSF] however, you must set up DEFBOO.CMD to boot from the system disk and perform the following procedure:

1. Press Ctrl/P to obtain the console prompt.
2. Enter the following BOOT commands:

```
>>> I/K
>>> BOOT/NOSTART
>>> DEPOSIT R5 F0000000
>>> CONTINUE
```

3 Assuming you set up DEFBOO.CMD properly, the system will boot from the SYSF directory on the system disk, and continue with the upgrade procedure.

If the system does not boot, enter the SHOW CONFIGURATION console command, and verify the DEFBOO.CMD contains the correct information for your configuration. For more information, see the *VMS Upgrade and Installation Supplement: VAX 9000 Series*.

B Files in the VMS Library and Optional Save Sets

This appendix lists the contents of the VMS *library* and *optional* save sets.

VMS Library

Accounting Log Report Generator Utility
Analyze Object File Utility (ANALYZE/OBJECT)
Debugger Utility (DEBUG)
Delta debugger
DUMP
EDT documentation
Error Log Report Generator Utility (ANALYZE/ERROR)
File Access Control List Utilities
Files–11 ODS1 ACP
Foreign terminal support
FORTRAN require files
Image Dump Utility (ANALYZE/IMAGE)
Incoming Remote File Access files
Incoming Remote Terminal files
Input Queue Symbiont (Card Reader)
LAT-11 terminal server (via Ethernet)
MACRO Assembler
MACRO Libraries
MAIL
Message Utility (MESSAGE)
Monitor Utility
National Character Set Utility (NCS)
Network support
Network Test files
Object and Shareable Image libraries
PHONE
Print and Batch Queue Utilities
Programming support
Remote Task Loading
RMS Analyze and FDL Editor Utilities (ANALYZE/RMS, ANALYZE/FDL)
RMS Journaling files
RUNOFF
SDL intermediary form of STARLET.MLB
Secure User's Environment
Standalone BACKUP
System Dump Analyzer Utility (ANALYZE/SYSTEM, ANALYZE/CRASH)
System programming support
System Symbol Table file (SYS.STB)
TECO editor
Terminal Fallback Facility
VAX-C object libraries

VAXTPU
VMS HELP library
VMS workstation support
VMS workstation device support

VMS Optional

BLISS require files
Example files
Files for VMS development only
Files used for kitting
LPA-11 support
Miscellaneous files
Miscellaneous Symbol Table files
System map
User Environment Test Package (UETP)

C DECwindows Software Components Table

Table C–1 contains the name of each VMS DECwindows software component, the directory where the component can be found, a brief description of what the component does, and one of the following three component classifications:

- ALL—Indicates that the component is part of all installations.
- PROG—Indicates the component is part of the programming environment.
- WS—Indicates that the component is part of the workstation device support.
- XMPL—Indicates that the component is part of the programming examples.

Table C–1 DECwindows Software Components

Directory	File	Purpose	Component Class
CDA$LIBRARY:	DEFSTYLE.DDIF	DDIF document style guide	PROG
DECW$EXAMPLES:	DEMO_BUILD.COM	Procedure to compile demo sources	XMPL
	ALLOBJS.H	Demo source	XMPL
	BITMAP.C	Demo source	XMPL
	BITMAP.EXE	Demo source	XMPL
	BTRAP.C	Broadcast message trapper program file	XMPL
	BTRAP.EXE	Broadcast message trapper image file	XMPL
	BTRAP.UID	Broadcast message trapper support file	XMPL
	BTRAP.UIL	Broadcast message trapper support file	XMPL
	CLOCK.DDIF	Paint example	ALL
	DECBURGER.ADA	UIL DRM Demo source	XMPL
	DECBURGER.C	UIL DRM Demo souce	XMPL
	DECBURGER.COM	UIL DRM Demo file	XMPL
	DECBURGER.EXE	UIL DRM Demo image	XMPL
	DECBURGER.FOR	UIL DRM Demo file	XMPL
	DECBURGER.UID	UIL DRM Demo file	XMPL

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	DECBURGER.UIL	UIL DRM Demo file	XMPL
	DECBURGER_DEFS.FOR	UIL DRM Demo file	XMPL
	DECW$TRANSPORT_ EXAMPLE.EXE	Demo image	XMPL
	DIALOG.C	Demo source	XMPL
	DWTVMSPUSHTEST.B32	XUI toolkit demo	XMPL
	DWTVMSPUSHTEST.EXE	Demo image	XMPL
	HELLOWORLD.ADA	XUI toolkit demo source	XMPL
	HELLOWORLD.C	XUI toolkit demo	XMPL
	HELLOWORLD.EXE	XUI toolkit demo image	XMPL
	HELLOWORLD.FOR	XUI toolkit demo	XMPL
	HELLOWORLD.UID	XUI toolkit demo file	XMPL
	HELLOWORLD.UIL	XUI toolkit demo file	XMPL
	ICO.C	Demo source	XMPL
	ICO.EXE	Demo image	XMPL
	ICOPAS.PAS	Demo source	XMPL
	OBJCUBE.H	Demo source	XMPL
	OBJICO.H	Demo source	XMPL
	OBJTETRA.H	Demo source	XMPL
	PLAID.C	Demo source	XMPL
	PLAID.EXE	Demo image	XMPL
	POLYINFO.H	Demo source	XMPL
	STRINGS.MAR	Low-level DECwindows utility routine	XMPL
	TESTVHIST.C	Histogram widget exerciser program file	XMPL
	TESTVHIST.EXE	Histogram widget exerciser image file	XMPL
	TESTVHIST.UIL	Histogram widget exerciser support file	XMPL
	TESTVHIST.UID	VList widget exerciser support file	XMPL
	TESTVLIST.C	VList widget exerciser program file	XMPL
	TESTVLIST.EXE	VList widget exerciser image file	XMPL
	TESTVLIST.UID	VList widget exerciser support file	XMPL

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	TESTVLIST.UIL	VList widget exerciser support file	XMPL
	TLIST.C	TList widget source file	XMPL
	TLIST.H	TList widget support file	XMPL
	TLIST.NOTE	TList widget support file	XMPL
	TLIST.UIL	TList widget support file	XMPL
	VFRAME.C	VFrame widget source file	XMPL
	VFRAME.H	VFrame widget support file	XMPL
	VFRAME.UIL	VFrame widget support file	XMPL
	VDRAGEXAMPLE.C	VDrag exerciser source file	XMPL
	VDRAGEXAMPLE.EXE	VDrag exerciser image file	XMPL
	VDRAG.C	VDrag exerciser source file	XMPL
	VDRAG.NOTE	VDrag exerciser support file	XMPL
	VHEADER.C	VHeader widget source file	XMPL
	VHEADER.H	VHeader widget support file	XMPL
	VHEADER.UIL	VHeader widget support file	XMPL
	VHIST.C	VHist widget source file	XMPL
	VHIST.H	VHist widget support file	XMPL
	VHIST.NOTE	VHist widget support file	XMPL
	VHIST.UIL	VHist widget support file	XMPL
	VLIST.C	VList widget source file	XMPL
	VLIST.H	VList widget support file	XMPL
	VLIST.NOTE	VList widget support file	XMPL
	VLIST.UIL	VList widget support file	XMPL
	XLIBINTRO.ADA	Demo source	XMPL
	XLIBINTRO.C	Demo source	XMPL
	XLIBINTRO.EXE	Demo image	XMPL
	XLIBINTRO.FOR	Demo source	XMPL
	XPORTEXAMPLEDEF.R32	Demo source	XMPL
	XPORT_EXAMPLE.B32	Demo source	XMPL
	XPORT_EXAMPLE_QUEUE.MAR	Demo source	XMPL
	XPORT_EXAMPLE_XFER.MAR	Demo source	XMPL
DECW$FONT:	See *VMS DECwindows Xlib Programming Volume*	DECwindows fonts	WS

(continued on next page)

DECwindows Software Components Table

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
DECW$INCLUDE:	COMPOBJ.H	Xlib and XUI toolkit programming include files	PROG
	COMPOBJP.H	Xlib and XUI toolkit programming include files	PROG
	COMPOSITE.H	Xlib and XUI toolkit programming include files	PROG
	COMPOSITEP.H	Xlib and XUI toolkit programming include files	PROG
	CONSTRAINT.H	Xlib and XUI toolkit programming include files	PROG
	CONSTRAINP.H	Xlib and XUI toolkit programming include files	PROG
	CONVERT.H	Xlib and XUI toolkit programming include files	PROG
	CORE.H	Xlib and XUI toolkit programming include files	PROG
	COREP.H	Xlib and XUI toolkit programming include files	PROG
	CSTEXTP.H	Xlib and XUI toolkit programming include files	PROG
	CURSORFONT.H	Xlib and XUI toolkit programming include files	PROG
	DECW$DWTDEF.UIL	UIL language include file	PROG
	DECWDWTAPPLPROG.H	Xlib and XUI toolkit programming include files	PROG
	DECWDWTAPPLPROG.UIL	XUI toolkit support	PROG
	DECWDWTWIDGETPROG.H	Xlib and XUI toolkit programming include files	PROG
	DECWMHINTS.H	Xlib and XUI toolkit programming include files	PROG
	DRMDECLS.H	Xlib and XUI toolkit programming include file	PROG
	DRMPUBLIC.H	Xlib and XUI toolkit programming include file	PROG
	DVR$CC_DEF.H	CDA Viewer support file	ALL
	DVR$DECW_DEF.H	CDA Viewer support file	ALL
	DVR$MSG.H	CDA Viewer support file	ALL
	DWTAPPL.H	Xlib and XUI toolkit programming include files	PROG

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	DWTAPPL.UIL	Xlib and XUI toolkit programming include files	PROG
	DWTWIDGET.H	Xlib and XUI toolkit programming include files	PROG
	EVENT.H	Xlib and XUI toolkit programming include files	PROG
	INTRINSIC.H	Xlib and XUI toolkit programming include files	PROG
	INTRINSICP.H	Xlib and XUI toolkit programming include files	PROG
	KEYSYM.H	Xlib and XUI toolkit programming include files	PROG
	KEYSYMDEF.H	Xlib and XUI toolkit programming include files	PROG
	OBJECT.H	Xlib and XUI toolkit programming include files	PROG
	OBJECTP.H	Xlib and XUI toolkit programming include files	PROG
	RECTOBJ.H	Xlib and XUI toolkit programming include files	PROG
	RECTOBJP.H	Xlib and XUI toolkit programming include files	PROG
	SELECTION.H	Xlib and XUI toolkit programming include files	PROG
	SHELL.H	Xlib and XUI toolkit programming include files	PROG
	SHELLP.H	Xlib and XUI toolkit programming include files	PROG
	STRINGDEFS.H	Xlib and XUI toolkit programming include files	PROG
	TRANSLATE.H	Xlib and XUI toolkit programming include files	PROG
	VENDOR.H	Xlib and XUI toolkit programming include files	PROG
	VENDORP.H	Xlib and XUI toolkit programming include files	PROG
	WINDOWOBJ.H	Xlib and XUI toolkit programming include files	PROG
	WINDOWOBJP.H	Xlib and XUI toolkit programming include files	PROG

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	X.H	Xlib and XUI toolkit programming include files	PROG
	XATOM.H	Xlib and XUI toolkit programming include files	PROG
	XLIB.H	Xlib and XUI toolkit programming include files	PROG
	XMD.H	Xlib and XUI toolkit programming include files	PROG
	XOS.H	Xlib and XUI toolkit programming include files	PROG
	XPROTO.H	Xlib and XUI toolkit programming include files	PROG
	XPROTOSTR.H	Xlib and XUI toolkit programming include files	PROG
	XRESOURCE.H	Xlib and XUI toolkit programming include files	PROG
	XUTIL.H	Xlib and XUI toolkit programming include files	PROG
DECW$SYSTEM_ DEFAULTS:	DDIF$VIEW.UID	CDA Viewer support file	ALL
	DDIF$VIEWWGT.UID	CDA Viewer support file	ALL
	DECW$BOOKREADER.DAT	Bookreader support file	ALL
	DECW$BOOKREADER.UID	Bookreader support file	ALL
	DECW$CALC.DAT	Calculator support file	ALL
	DECW$CALC.UID	Calculator support file	ALL
	DECW$CALENDAR.DAT	Calendar support file	ALL
	DECW$CALENDAR.UID	Calendar support file	ALL
	DECW$CARDFILER.DAT	Cardfiler support file	ALL
	DECW$CARDFILER.UID	Cardfiler support file	ALL
	DECW$CLOCK.DAT	Clock support file	ALL
	DECW$CLOCK.UID	Clock support file	ALL
	DECW$LOGIN.DAT	Login support file	ALL
	DECW$LOGIN.UID	Login support file	ALL
	DECW$MAIL.DAT	DECwindows VMSmail support file	ALL
	DECW$MAIL_MAIN.UID	DECwindows VMSmail support file	ALL

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	DECW$MAIL_MISC.UID	DECwindows VMSmail support file	ALL
	DECW$MAIL_READ.UID	DECwindows VMSmail support file	ALL
	DECW$MAIL_SEND.UID	DECwindows VMSmail support file	ALL
	DECW$NOTEPAD.DAT	NOTEPAD support file	ALL
	DECW$NOTEPAD.UID	NOTEPAD support file	ALL
	DECW$PAINT.DAT	DECwindows Paint support file	ALL
	DECW$PAINT.UID	DECwindows Paint support file	ALL
	DECW$PRINTWGT.UID	Print support file	ALL
	DECW$PRINTWGT_UI.UID	Print support file	ALL
	DECW$PUZZLE.DAT	Puzzle support file	ALL
	DECW$PUZZLE.UID	Puzzle support file	ALL
	DECW$SESSION.DAT	Session manager support file	ALL
	DECW$SESSION.UID	Session manager support file	ALL
	DECW$SM_BW.DAT	Session manager support file	ALL
	DECW$SM_COLOR.DAT	Session manager support file	ALL
	DECW$SM_GENERAL.DAT	Session manager support file	ALL
	DECW$SM_GRAY.DAT	Session manager support file	ALL
	DECW$TERMINAL.DAT	Terminal support file	ALL
	DECW$TERMINAL.UID	Terminal support file	ALL
	DECW$WINMGR.DAT	Window manager support file	ALL
	DECW$WINMGR.UID	Window manager support file	ALL
	VUE$MASTER.UID	FileView defaults file	ALL
	XNL_88591.UID	XNLS ISOLATIN1 character set	PROG
	XNL_88598.UID	XNLS ISOLATIN8 character set	PROG
	XNL_DE_CH.UID	XNLS Swiss German culture file	PROG
	XNL_DE_DE.UID	XNLS German culture file	PROG
	XNL_EN_AU.UID	XNLS Australian culture file	PROG
	XNL_EN_GB.UID	XNLS British English culture file	PROG

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	XNL_EN_NZ.UID	XNLS New Zealand culture file	PROG
	XNL_EN_PG.UID	XNLS Papua New Guinea culture file	PROG
	XNL_EN_US.UID	XNLS American English culture file	PROG
	XNL_ES_ES.UID	XNLS Spanish culture file	PROG
	XNL_FI_FI.UID	XNLS Finnish culture file	PROG
	XNL_FJ_FJ.UID	XNLS Fiji culture file	PROG
	XNL_FR_BE.UID	XNLS Belgian French culture file	PROG
	XNL_FR_CA.UID	XNLS Canadian French culture file	PROG
	XNL_FR_CH.UID	XNLS Swiss French culture file	PROG
	XNL_FR_FR.UID	XNLS French culture file	PROG
	XNL_IT_CH.UID	XNLS Swiss Italian culture file	PROG
	XNL_IT_IT.UID	XNLS Italian culture file	PROG
	XNL_IW_IL.UID	XNLS Hebrew culture file	PROG
	XNL_NL_BE.UID	XNLS Flemish culture file	PROG
	XNL_NL_NL.UID	XNLS Dutch culture file	PROG
	XNL_NO_NO.UID	XNLS Norwegian culture file	PROG
	XNL_SV_SE.UID	XNLS Swedish culture file	PROG
SYS$LIBRARY:	CDA$ACCESS.EXE	The DDIF toolkit and DDIF converter toolkit Run-time-library shareable image	PROG
	CDA$CDA_.ADA	CDA Ada binding	PROG
	CDA$DEF.*†	DDIF converter language binding	PROG
	CDA$DTIF_TO_DDIF.EXE	CDA DTIF to DDIF converter image	ALL
	CDA$MSG.*†	DDIF message symbols	PROG
	CDA$WRITE_ANALYSIS.EXE	CDA image	ALL
	DDIF$DDIF_.ADA	DDIF Ada binding	PROG
	DDIF$DEF.*†	DDIF language binding	PROG

†BASIC, BLISS, FORTRAN, C, Macro, Pascal, and PLI languages are supported. Asterisk (*) represents the appropriate file extension for each language.

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	DDIF$READ_TEXT.EXE	Text front end shareable image	ALL
	DDIF$VIEWSHR.EXE	DDIF viewer shareable image	ALL
	DDIF$WRITE_PS.EXE	Text image	ALL
	DDIF$WRITE_TEXT.EXE	Text back end shareable image	ALL
	DECW$AILSHR.EXE	Image support shareable image	ALL
	DECW$CALENDAR_PROLOG.PS	Calendar support file	ALL
	DECW$CURSOR.H	Cursor include files	PROG
	DECW$DRIVER.MLB	Driver macro library	PROG
	DECW$DWT_.ADA	Toolkit Ada binding	PROG
	DECW$DWTDEF.*†	Widget include files	PROG
	DECW$DWTENTRY.*		
	DECW$DWTLIBSHR.EXE	XUI toolkit shareable image	PROG
	DECW$DWTMSG.*	Toolkit message file	PROG
	DECW$DWTSTRUCT.*	Widget include files	PROG
	DECW$DWTWIDGETSTRUCT.*	Widget include files	PROG
	DECW$DWTWIDGETDEF.*†	Widget include files	PROG
	DECW$LOGINOUT.EXE	DECwindows extension to VMS loginout image	ALL
	DECW$MAILSHR.EXE	DECwindows VMSmail shareable image	ALL
	DECW$PRINTWGT.UID	Print support file	ALL
	DECW$PRINTWGTSHR.EXE	DECwindows print widget shareable image	ALL
	DECW$SERVER_DDX_GA.EXE	Server color device support shareable image	WS
	DECW$SERVER_DDX_GB.EXE	Server color device support shareable image	WS
	DECW$SERVER_DDX_GC.EXE	Server monochrome device support shareable image	WS
	DECW$SERVER_DDX_GE.EXE	Server scanproc device-dependent shareable image	WS
	DECW$SERVER_DIX.EXE	Server device independent support	WS

†BASIC, BLISS, FORTRAN, C, Macro, Pascal, and PLI languages are supported. Asterisk (*) represents the appropriate file extension for each language.

(continued on next page)

DECwindows Software Components Table

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	DECW$SERVER_EXTENSION_ PEX.EXE	PHIGS X Server Extension	WS
	DECW$SVEXT_ADOBE_DPS_ EXTENSION.EXE	Display Postscript server extension image	WS
	DECW$TERMINALSHR.EXE	Terminal shareable image	ALL
	DECW$TRANSPORT_ COMMON.EXE	Common transport shareable image	ALL
	DECW$TRANSPORT_DECNET.EXE	DECnet transport image	ALL
	DECW$TRANSPORT_LAT.EXE	LAT transport image	ALL
	DECW$TRANSPORT_LOCAL.EXE	Local transport image	ALL
	DECW$TRANSPORT_TCPIP.EXE	TCP/IP transport image	ALL
	DECW$UIL.ENV	LSE template for UIL compilier	PROG
	DECW$X_.ADA	Xlib Ada binding	PROG
	DECW$XLIBDEF.*†	Language Xlib binding	PROG
	DECW$XLIBMSG.*†	Language message symbols	PROG
	DECW$XLIBSHR.EXE	Xlib shareable image	PROG
	DECW$XPORTCOM.H	Common transport definitions	PROG
	DECW$XPORTCOM.MAR	Common transport definitions	PROG
	DECW$XPORTCOM.R32	Common transport definitions	PROG
	DECW$XPORTDEF.H	Transport definitions	PROG
	DECW$XPORTDEF.MAR	Transport definitions	PROG
	DECW$XPORTDEF.R32	Transport definitions	PROG
	DECW$XPORTMAC.R32	Transport BLISS macros	PROG
	DECW$XPORTMSG.R32	Transport message symbols	PROG
	PHIGS$GB_UCODE.EXE	PHIGS image	WS
	XDPS$DPSBINDINGSSHR.EXE	Display Postscript VAX Bindings image	ALL
	XDPS$DPSCLIENTSHR.EXE	Display Postscript client library image	ALL
	XDPS$DPSLIBSHR.EXE	Display Postscript xlib extension image	ALL
	XDPS$DPSOPS.*‡	Display Postscript include file	PROG
	XDPS$DPSXCLIENT.*‡	Display Postscript include file	PROG
	XDPS$MASTERDPSVM.DAT	Display Postscript support file	ALL

†BASIC, BLISS, FORTRAN, C, Macro, Pascal, and PLI languages are supported. Asterisk (*) represents the appropriate file extension for each language.

‡BASIC, BLISS, FORTRAN, C, and Macro languages are supported. Asterisk (*) represents the appropriate file extension for each language.

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	XDPS$PSOPS.*‡	Display Postscript include file	PROG
	XNL$SHR.EXE	XNLS shareable image	ALL
	XDPS$XDPS.*‡	Display Postscript include file	PROG
	XDPS$XDPSLIB.*‡	Display Postscript include file	PROG
	XNL$DEF.*‡	XNLS include files	PROG
	XNL$MSG.*‡	XNLS message symbols	PROG
SYS$MANAGER:	DECW$CHECK_PARAMS.COM	Check AUTOGEN parameters	ALL
	DECW$DEVICE.COM	Device detection and configuration	ALL
	DECW$DEVICE_GE.COM	Scanproc device detection and configuration	WS
	DECW$LOGICALS.COM	Logical name table initialization and start	ALL
	DECW$PRIVATE_SERVER_ SETUP.TEMPLATE	Server customization template	WS
	DECW$RGB.COM	Definitions of the color names used by the XStoreNamedColor and XAllocNamedColor requests	WS
	DECW$STARTAPPS.COM	User environment	ALL
	DECW$STARTLIBS.COM	Programming environment startup	PROG
	DECW$STARTSERVER.COM	Server initialization and startup	WS
	DECW$STARTSM.COM	Starts the session manager	ALL
	DECW$STARTUP.COM	Starts the DECwindows environment	ALL
	DECW$SYLOGIN.TEMPLATE	System login file template	ALL
SYS$MESSAGE:	DDIF$VIEWMSG.EXE	Viewer message file	ALL
	DECW$DWTERRDB.DAT	Toolkit error support file	PROG
	DECW$DWTMSG.EXE	Toolkit message file	PROG
	DECW$TERMINALMSG.EXE	DECterm message file	ALL
	DECW$TRANSPORTMSG.EXE	Transport message image	ALL
	DECW$XLIBERRDB.DAT	Xlib resource database	PROG

‡BASIC, BLISS, FORTRAN, C, and Macro languages are supported. Asterisk (*) represents the appropriate file extension for each language.

(continued on next page)

DECwindows Software Components Table

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	DECW$XLIBMSG.EXE	Xlib message image	PROG
SYS$SYSTEM:	CDA$CONVERT.EXE	DDIF converter image	ALL
	DDIF$VIEW.EXE	DDIF viewer	ALL
	DECW$BOOKREADER.EXE	Bookreader image	ALL
	DECW$CALC.EXE	Calculator image	ALL
	DECW$CALENDAR.EXE	Calendar image	ALL
	DECW$CARDFILER.EXE	Cardfiler image	ALL
	DECW$CLOCK.EXE	Clock image	ALL
	DECW$FONTCOMPILER.EXE	Font compiler image	ALL
	DECW$MAIL.EXE	DECwindows VMSmail image	ALL
	DECW$NOTEPAD.EXE	Notepad image	ALL
	DECW$PAINT.EXE	Paint image	ALL
	DECW$PUZZLE.EXE	Puzzle image	ALL
	DECW$SERVER_MAIN.EXE	Server main image	WS
	DECW$SESSION.EXE	Manages the session and contains the control panel	ALL
	DECW$SETSHODIS.EXE	SET/SHOW DISPLAY command image	PROG
	DECW$STARTLOGIN.EXE	Starts the DECwindows LOGINOUT.EXE image	ALL
	DECW$TERMINAL.EXE	Terminal emulator controller	ALL
	DECW$TERMINAL_CREATE.EXE	DECterm image	ALL
	DECW$UILCOMPILER.EXE	UIL compiler	PROG
	DECW$WINMGR.EXE	DECwindows window manager	ALL
	VUE$MASTER.EXE	FileView image	ALL
	XDPS$PSWRAP.EXE	Display Postscript PSWRAP utility image	PROG
SYS$LOADABLE_ IMAGES:	GAADRIVER.EXE	VAXstation II/GPX driver	WS
	GABDRIVER.EXE	VAXstation 2000/GPX driver	WS
	GBBDRIVER.EXE	VAXstation 35x0 driver	WS
	GCADRIVER.EXE	VAXstation II monochrome device driver	WS
	GCBDRIVER.EXE	VAXstation 2000 monochrome device driver	WS

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	GEBDRIVER.EXE	scanproc graphics device driver	WS
	IKDRIVER.EXE	Keyboard decoder driver	WS
	IMDRIVER.EXE	Mouse decoder driver	WS
	INDRIVER.EXE	Common input function driver	WS
	PYDRIVER.EXE	Pseudo terminal driver	ALL
	TWDRIVER.EXE	Pseudo terminal driver	ALL
	WSDRIVER.EXE	Set display driver	ALL
DECW$BOOK:	BOOKREADER.DECW$BOOK	Online user's guide for the Bookreader	ALL
	LIBRARY.DECW$BOOKSHELF	Bookreader bookshelf	ALL
DECW$KEYMAP:	*.DECW$KEYMAP	DECwindows Keymap files	WS
SYS$HELP:	DDIF$VIEW.HLB	View help file	ALL
	DECW$BOOKREADER.HLB	Bookreader help file	ALL
	DECW$CALC.HLB	Calculator help file	ALL
	DECW$CALENDAR.HLB	Calendar help file	ALL
	DECW$CARDFILER.HLB	Cardfiler help file	ALL
	DECW$CLOCK.HLB	Clock help file	ALL
	DECW$HELPHELP.HLB	HELP help file	PROG
	DECW$MAIL.HLB	Mail help file	ALL
	DECW$NOTEPAD.HLB	Notepad help file	ALL
	DECW$PAINT.HLB	Paint help file	ALL
	DECW$PRINTWGT.HLB	Print help file	ALL
	DECW$PUZZLE.HLB	Puzzle help file	ALL
	DECW$SESSION.HLB	Session Manager help file	ALL
	DECW$TERMINAL.HLB	Terminal Emulator help file	ALL
	DECW$VUE.HLB	FileView help file	ALL
SYS$UPDATE:	DECW$COMPILE_ADA_UNITS.COM	Ada build file	PROG
	DECW$KITBLD.DAT	Installation support file	ALL
	DECW$KITBLD.IDX	Installation support file	ALL
	DECW$TAILOR.EXE	Tailoring program	ALL

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	DECW$TAILOR_ON.TEMPLATE	Tailoring support file	ALL
VUE$LIBRARY:	VUE$*.COM	FileView verb command files	ALL
	VUE$SYSTEM_PROFILE.VUE$DAT	FileView default system profile	ALL
XDPS$EXAMPLES:	ACTIONPRCS.H	Display Postscript PSDRAW example application file	XMPL
	ACTIONS.C	Display Postscript PSDRAW example application file	XMPL
	ACTIONS.H	Display Postscript PSDRAW example application file	XMPL
	BIGICON.BIT	Display Postscript PSDRAW example application file	XMPL
	BUILD_PSDRAW.COM	Display Postscript PSDRAW example application file	XMPL
	BUTTON.C	Display Postscript PSDRAW example application file	XMPL
	BUTTON.H	Display Postscript PSDRAW example application file	XMPL
	BUTTONP.H	Display Postscript PSDRAW example application file	XMPL
	COLOR.C	Display Postscript PSDRAW example application file	XMPL
	COLOR.H	Display Postscript PSDRAW example application file	XMPL
	COORDS.C	Display Postscript PSDRAW example application file	XMPL
	COORDS.H	Display Postscript PSDRAW example application file	XMPL
	COORDSP.H	Display Postscript PSDRAW example application file	XMPL
	DRAWING.C	Display Postscript PSDRAW example application file	XMPL
	DRAWING.H	Display Postscript PSDRAW example application file	XMPL
	DRAWINGP.H	Display Postscript PSDRAW example application file	XMPL
	KNOBS.C	Display Postscript PSDRAW example application file	XMPL

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	KNOBS.H	Display Postscript PSDRAW example application file	XMPL
	KNOBSP.H	Display Postscript PSDRAW example application file	XMPL
	MENUS.C	Display Postscript PSDRAW example application file	XMPL
	MENUS.H	Display Postscript PSDRAW example application file	XMPL
	MODE.C	Display Postscript PSDRAW example application file	XMPL
	MODE.H	Display Postscript PSDRAW example application file	XMPL
	MODEP.H	Display Postscript PSDRAW example application file	XMPL
	M_PI.H	Display Postscript PSDRAW example application file	XMPL
	OPS.C	Display Postscript PSDRAW example application file	XMPL
	PANE.C	Display Postscript PSDRAW example application file	XMPL
	PANE.H	Display Postscript PSDRAW example application file	XMPL
	PANEP.H	Display Postscript PSDRAW example application file	XMPL
	PIXMGR.C	Display Postscript PSDRAW example application file	XMPL
	PIXMGR.H	Display Postscript PSDRAW example application file	XMPL
	POPUP.C	Display Postscript PSDRAW example application file	XMPL
	POPUP.H	Display Postscript PSDRAW example application file	XMPL
	PROP.C	Display Postscript PSDRAW example application file	XMPL
	PROP.H	Display Postscript PSDRAW example application file	XMPL
	PROPP.H	Display Postscript PSDRAW example application file	XMPL
	PSDRAW.C	Display Postscript PSDRAW example application file	XMPL

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	PSDRAW.COM	Display Postscript PSDRAW example application file	XMPL
	PSDRAW.DAT	Display Postscript PSDRAW example application file	XMPL
	PSDRAW.H	Display Postscript PSDRAW example application file	XMPL
	PSDRAWCOMPLIB.DAT	Display Postscript PSDRAW example application file	XMPL
	PSDRAWKINDS.DAT	Display Postscript PSDRAW example application file	XMPL
	PSWINDOW.C	Display Postscript PSDRAW example application file	XMPL
	PSWINDOW.H	Display Postscript PSDRAW example application file	XMPL
	PSWINDOWP.H	Display Postscript PSDRAW example application file	XMPL
	PSWINDW.C	Display Postscript PSDRAW example application file	XMPL
	PSWINDW.PSW	Display Postscript PSDRAW example application file	XMPL
	RECT.C	Display Postscript PSDRAW example application file	XMPL
	RECT.H	Display Postscript PSDRAW example application file	XMPL
	RECTP.H	Display Postscript PSDRAW example application file	XMPL
	ROUND.C	Display Postscript PSDRAW example application file	XMPL
	SCRN.C	Display Postscript PSDRAW example application file	XMPL
	SCRN.H	Display Postscript PSDRAW example application file	XMPL
	SCRNP.H	Display Postscript PSDRAW example application file	XMPL
	SCROLLW.C	Display Postscript PSDRAW example application file	XMPL
	SCROLLW.H	Display Postscript PSDRAW example application file	XMPL
	SCROLLWP.H	Display Postscript PSDRAW example application file	XMPL

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	SMALLICON.BIT	Display Postscript PSDRAW example application file	XMPL
	STRING.H	Display Postscript PSDRAW example application file	XMPL
	STRINGS.MAR	Display Postscript PSDRAW example application file	XMPL
	UTIL.C	Display Postscript PSDRAW example application file	XMPL
	UTIL.H	Display Postscript PSDRAW example application file	XMPL
	VALUE.C	Display Postscript PSDRAW example application file	XMPL
	VALUE.H	Display Postscript PSDRAW example application file	XMPL
	VALUEP.H	Display Postscript PSDRAW example application file	XMPL
	WIDGET.C	Display Postscript PSDRAW example application file	XMPL
	WIDGET.H	Display Postscript PSDRAW example application file	XMPL
	WIDGETINFO.C	Display Postscript PSDRAW example application file	XMPL
	WIDGETINFO.H	Display Postscript PSDRAW example application file	XMPL
	WRAPS.C	Display Postscript PSDRAW example application file	XMPL
	WRAPS.PSW	Display Postscript PSDRAW example application file	XMPL
	XMAPRGBTOPIXEL.C	Display Postscript PSDRAW example application file	XMPL
	PSDRAW.EXE	Display Postscript PSDRAW example application file	XMPL
	PSDRAW_README.PS	Display Postscript PSDRAW example application file	XMPL
XDPS$INCLUDE:	DPSCLIENT.H	Display Postscript support file	PROG
	DPSCUSTOMOPS.H	Display Postscript support file	PROG
	DPSEXCEPT.H	Display Postscript support file	PROG
	DPSFRIENDS.H	Display Postscript support file	PROG
	DPSOPS.H	Display Postscript support file	PROG

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	DPSXCLIENT.H	Display Postscript support file	PROG
	PSCUSTOMOPS.H	Display Postscript support file	PROG
	PSOPS.H	Display Postscript support file	PROG
	XDPS.H	Display Postscript support file	PROG
	XDPSLIB.H	Display Postscript support file	PROG

D Layered Products Caution

Because of the way the VMS Version 5.4 upgrade procedure is designed, you should not have to re-install most layered products after the upgrade. If a product is available (refer to Table D–1), yet exhibits unexpected behavior once Version 5.4 is running, check the *Upgrade Notes* section in Chapter 3 of the *VMS Version 5.4 Release Notes* for layered product restrictions. If problems persist, contact your Digital support representative.

Table D–1 lists the most recent versions of layered products that are supported for VMS Version 5.4 as of June 19, 1990. As this list is subject to change, Digital recommends that you refer to the System Support Addendum and Release Notes of each layered product for the latest information. If problems persist, contact your Digital support representative.

Table D–1 Layered Products Currently Available for VMS Version 5.4

Product Name	Most Recent Version Number
3270 Terminal Option Software	1.1
AAF01/VMS Subroutine Library	2.0
ADF01/VMS Subroutine Library	4.0
ALL-IN-1	2.3
ALL-IN-1	2.4[1]
All-in-1 Mail	1.0
All-in-1 PC Server For VMS	1.0
ALL-IN-1 Starter	1.0
ALL-IN-1 System for Sales and Marketing	1.2
CDA Converter Library for VMS	1.1
CMR21 Host Utility	1.1
DEC Capacity Planner for VMS	1.1
DEC Extended Basic Mode (XBM) CO3 Access For VMS	4.0
DEC Extended Basic Mode (XBM) CO3 Gateway For VMS	4.0
DEC GKS for VMS	4.1
DEC IEZ11 Class Driver For VMS	1.0
DEC PHIGS for VMS	2.1
DEC TM32 Software for VMS	1.0
DECdecision	1.1

[1]See *VMS Version 5.4 Release Notes* for additional information.

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.4

Product Name	Most Recent Version Number
DECdx/VMS	1.2
DECforms	1.2
DECimage Application Services for VMS	2.0
DECimage Scan Software for VMS	2.0
DECimage Storage Manager for VMS	1.0
DECintact	1.1
Decision Expert for VMS	1.0
DECLAB for VMS	1.0
DECMCC Enterprise Management Station	1.0
DECMCC Site Management Station	1.0
DECnet Router Server	1.2
DECnet/SNA Data Transfer Facility	2.0
DECnet/SNA Gateway	1.5
DECnet/SNA Gateway for Channel Transport	1.0
DECnet/SNA Gateway for Synchronous Transport	1.1
DECnet/SNA VMS 3270 Data Stream Programming Interface	1.4
DECnet/SNA VMS 3270 Terminal Emulator	1.5
DECnet/SNA VMS APPC/LU6.2 Programming Interface	2.1
DECnet/SNA VMS Application Programming Interface	2.3
DECnet/SNA VMS DISOSS Document Exchange Facility	1.4
DECnet/SNA VMS Distributed Host Command Facility	1.2
DECnet/SNA VMS Gateway Management	2.0
DECnet/SNA VMS Printer Emulator	1.2
DECnet/SNA VMS Remote Job Entry	1.4
DECnet–VAX	5.4
DECpage	3.1
DECprint Utility for Postscript to Sixel Printing for VMS	1.0
DECrouter 200	1.1
DECrouter 250	1.0
DECrouter 2000	1.2
DECscheduler for VMS	1.0
DECserver 100 for VMS and MicroVMS	2.0
DECserver 200 for VAX/VMS and MicroVMS	3.0
DECserver 250 for VMS	1.0
DECserver 300 for VMS	1.0
DECserver 500 for VAX/VMS and MicroVMS	2.0

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.4

Product Name	Most Recent Version Number
DECtalk Mail Access	1.1
DECtrace For VMS	1.0
DECview3D for VMS	1.0
DECvoice Software	1.1
DECwindows DECnet/SNA 3270 Terminal Emulator for VMS	1.0
DECwrite for VMS	1.0
DECxpress 100 VMS Client Software	2.0
DEC/EDI	1.0
EDCS II	2.0
Ethernet Terminal Server for VMS and MicroVMS	3.0
External Document Exchange with IBM DISOSS	2.1
Fortran IV/VAX to RSX	2.8
IEX-VMS-Driver	4.1
Internet Portal	1.1
KMV1A MicroVAX Driver	2.1
KMV1A MicroVAX Driver and Development Tools	2.1
KMV1A MicroVAX Driver and X.25 Link Level Software	2.1
LAN Traffic Monitor for VMS	1.2
LCG01 Software	1.5
MicroVAX MIRA Switch Control	2.1
MicroVAX VSV21 Support Software	3.0
MicroVAX/DRQ3B Device Driver	1.2
MIRA High Availability Management Software For VMS	1.0
MUXserver 100 Remote Terminal Server	2.3
MUXserver 300 Remote Terminal Server for VMS	1.1
NMCC/DECnet Monitor	2.2
NMCC/VAX ETHERnim	2.2
PBXserver	2.1
PCLAN/Server VMS	3.0
PDP–11 C for VMS	1.0
PDP–11 DATATRIEVE/VAX	3.3
PDP–11 FORTRAN–77/VAX to RSX	5.4
PDP–11 Symbolic Debugger/VAX to RSX	2.1
Remote Bridge Management Software	2.0
Remote System Manager	2.2
Session Support Utility	1.1

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.4

Product Name	Most Recent Version Number
Terminal Server Manager	1.3
ULTRIX Mail Connection	1.1
VAX 2780/3780 Protocol Emulator	1.7
VAX 3271 Protocol Emulator	2.5
VAX ACMS	3.1
VAX Ada	2.1
VAX ADE	2.4
VAX APL	3.2
VAX BASIC	3.4
VAX BLISS-32 Implementation Language	4.6
VAX C	3.1
VAX CDD/Plus	4.2
VAX COBOL	4.3
VAX COBOL Generator	1.3
VAX Computer Integrated Telephony Applications Interface	2.0
VAX Computer Integrated Telephony Message Desk	1.0
VAX Computer Integrated Telephony Server	2.0
VAX CoProcessor/RSX	1.1
VAX Data Distributor	2.1
VAX DATATRIEVE	5.0
VAX DBMS	4.2
VAX DEC/CMS	3.3
VAX DEC/MAP	3.0
VAX DEC/MMS	2.5
VAX DEC/Shell	2.2
VAX DEC/Test Manager	3.1
VAX DECalc	3.1
VAX DECalc-Plus	3.1
VAX DECalc/DECgraph Package	3.1
VAX DECgraph	1.5
VAX DECrad	4.0
VAX DECscan VMS and ELN Bitbus Software Drivers	2.0
VAX DECscan VMS Software Toolkit	2.1
VAX DECslide	1.3
VAX DECspell Verifier/Corrector	1.1
VAX DIBOL	4.1

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.4

Product Name	Most Recent Version Number
VAX Disk Striping Driver for VMS	1.0
VAX Distributed File Service	1.2[1]
VAX Distributed Name Service	1.1
VAX Distributed Queuing Service	1.1
VAX Document	1.2
VAX DSM	5.2
VAX DT07	3.0
VAX DY32	3.0
VAX Encryption	1.2
VAX FMS	2.4
VAX FORTRAN	5.4
VAX FTAM	1.1
VAX Grammar Checker	1.0
VAX ISDN	1.0
VAX ISDN Access	1.0
VAX Jukebox Control Software	1.0
VAX KCT32	2.0
VAX Key Distribution Center	1.2
VAX KMS11-BD/BE HDLC/BSC Framing Software	2.0
VAX KMS11-BD/BE X.25 Link Level Software	2.0
VAX Language Sensitive Editor	3.0
VAX LISP/VMS	3.1
VAX LN03 Image Support Software	1.0
VAX Mailgate for MCI Mail	2.0
VAX Media Manager	1.0
VAX Message Router	3.1
VAX Message Router/P Gateway	1.1
VAX Message Router/S Gateway	1.1
VAX Message Router Telex Gateway	1.0
VAX Message Router X.400 Gateway	2.1
VAX Notes	2.1
VAX OPS5	3.0
VAX OSI Application Kernel	1.1
VAX Packetnet System Interface	4.3
VAX Packetnet System Interface Access	4.3
VAX PASCAL	4.0

[1]See *VMS Version 5.4 Release Notes* for additional information.

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.4

Product Name	Most Recent Version Number
VAX PCL	2.0
VAX Performance Advisor	2.1
VAX Performance and Coverage Analyzer	2.2
VAX PL/I	3.3
VAX PrintServer Client Software	3.0
VAX PrintServer Supporting Host Software	3.2
VAX Public Access Communication	1.2
VAX Rally	2.1
VAX Rdb/ELN	2.2
VAX Rdb/VMS	3.1
VAX Real-Time Accelerator Software	2.0
VAX ReGIS to Sixels Converter	1.0
VAX Remote Environmental Monitoring Software	1.2
VAX RMS Journaling	5.4
VAX Scan	1.2
VAX ScriptPrinter Software	2.1
VAX Software Performance Monitor	3.4
VAX Software Project Manager	1.2
VAX Source Code Analyzer	2.1
VAX Storage Library System	2.0
VAX TDMS	1.9
VAX TEAMDATA	1.4
VAX TU70/72 Device Driver	1.2[1]
VAX Volume Shadowing	5.4
VAX Vertical Forms Printing for VMS	1.0
VAX VTX	4.1
VAX Wide Area Network Device Drivers	1.1
VAX XWAY	1.2
VAX–11 RSX	2.4
VAXcluster Console System	1.2
VAXcluster Software	5.4
VAXELN Ada	2.0
VAXELN Toolkit	4.1
VAXlink for IMS and VSAM Client	1.1
VAXset	9.0
VIDA for DB2	1.0

[1]See *VMS Version 5.4 Release Notes* for additional information.

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.4

Product Name	Most Recent Version Number
VMS Services for PCS	3.1
VMS Workstation Software	4.3[1]
VMS/SNA	2.0
VMS/ULTRIX Connection (tm)	1.3
WPS-PLUS/VMS	3.1
X25Portal 2000	1.0
X25Router 2000	1.1

[1]See *VMS Version 5.4 Release Notes* for additional information.

Glossary

boot or bootstrap: The process of loading system software into a processor's main memory. This guide uses the term *boot* to refer to this process.

boot command procedure: A program stored on the console fixed disk that is used to boot the VMS operating system from a specified controller. Digital provides a boot command procedure for each controller that the processor supports.

boot name: The abbreviated name of the boot command procedure you use to boot the system.

boot server: A VAX computer that is part of a local area VAXcluster. The boot server in a local area VAXcluster has a system disk that contains cluster common files; other nodes in the cluster (satellite nodes) can access these files. See also *satellite node*.

CI-only VAXcluster: A computer system consisting of a number of VAX computers. It uses only the computer interconnect (CI) to communicate with other VAX computers in the cluster. These computers share a single file system.

computer interconnect: A computer interconnect (CI) is a type of I/O subsystem. It links VAX computers to each other and to HSC devices.

DECW$TAILOR: A software program that lets you add or remove DECwindows files from your system disk.

device name: The name you use to identify a device on the system. A device name indicates the device code, controller designation, and unit number.

Hierarchical Storage Controller (HSC) device: A self-contained, intelligent, mass storage subsystem that lets VAX computers in a VAXcluster environment share disks. Examples of HSC devices are the HSC50 and the HSC70.

HSC drive: Any disk or tape drive that is connected to an HSC device is referred to as an HSC drive. A system disk on an HSC drive can be shared by several VAX computers in a VAXcluster environment.

local area VAXcluster: Consists of a VAX computer that acts as a boot server and a number of low-end VAX computers that act as satellite nodes. Ethernet connects all of the computers. These computers share a single file system.

local drive: Any drive that is connected directly to a VAX computer is referred to as a local drive.

media: A generic term that refers to any packaging agent capable of storing computer software. Examples of media are compact discs, magnetic tapes, floppy diskettes, disk packs, tape cartridges, etc.

Glossary

mixed-interconnect VAXcluster: A computer system consisting of a number of VAX computers. It uses both the computer interconnect (CI) and Ethernet to communicate with other VAX computers in the cluster. These computers share a single file system.

satellite node: A VAX computer that is part of a local area VAXcluster. A satellite node is booted remotely from the system disk of the boot server in the local area VAXcluster. See also *boot server*.

save set: The format that the Backup Utility stores files in. The VMS operating system is shipped in this format.

scratch disk: A blank disk or a disk with files that you no longer need.

source drive: When performing an upgrade or installation, the source drive is the drive that holds the distribution kit.

spin up/spin down: To spin up means to bring a disk drive up to operating speed. To spin down means to bring it to a gradual stop.

standalone BACKUP: A subset of the Backup Utility that is booted into main memory instead of running under the control of the VMS operating system.

standalone system: A computer system with only one VAX computer.

system disk: The disk that contains (or will contain) the VMS operating system. A VMS system disk is set up so that most of the VMS files can be shared by several VAX computers. In addition, each computer has its own directory on the system disk that contains its page, swap, and dump files.

target drive: When performing an upgrade or installation, the target drive is the drive that holds the system disk.

VAXcluster environment: A computer system consisting of a number of VAX computers. There are three types of VAXcluster environments: CI-only, local area, and mixed-interconnect.

VMS User Environment Test Package (UETP): A software package that tests all the standard peripheral devices on your system, various commands and operating system functions, the system's multi-user capability, DECnet–VAX, and the VAXcluster environment.

VMSTAILOR: A software program that lets you add or remove optional VMS files from your system disk.

Index

A

Account
  disabling during installation • 3–13
  setting up after an installation • 4–2
Accounting logs
  effect of upgrade on • 5–3
Ada
  compiling source files • 11–10
  programming interfaces • 11–10
AGEN$PARAMS.REPORT (file created by AUTOGEN) • 4–6
Allocation class
  See also ALLOCLASS parameter
  determining • 2–4
ALLOCLASS parameter • 2–4
  supplying during VAXcluster installation • 3–11
ANALYZE/DISK_STRUCTURE utility
  error message • 6–5
  using before an upgrade • 6–5
Applications files for VMS DECwindows • 8–9
Applying the mandatory update • 8–13
AUTOGEN • 4–6
  feedback data • 6–2
  running after an installation • 4–6
  running after an upgrade • 9–13
  running at end of installation • 3–16
  running at end of upgrade • 8–15
  running before an upgrade • 6–2
AUTOGEN.PAR
  creation of • 8–13
Automatic restart
  for upgrade • 6–7

B

Backing up the system disk • 4–6, 9–13
BACKUP command
  using to create a system disk • 3–3
Backup procedures
  before an upgrade • 6–3
Boot command procedures
  See DEFBOO and dddGEN
Booting
  manually during upgrade procedure • 8–6
Bugcheck message
  during UETP • 10–29
Building standalone BACKUP • 4–2

C

Checklist
  post-installation • 4–1
  post-upgrade • 9–1
  pre-installation • 2–1
  pre-upgrade • 6–1
Choosing a windowing system • 3–12
CI-only VAXcluster
  information needed to install • 2–3
  installation questions • 3–10
Cluster
  See VAXcluster
Cluster group number
  rules for creating • 2–4
Cluster password
  rules for creating • 2–4
Cluster upgrade
  See Concurrent upgrade
  See Rolling upgrade
CLUSTER_SIZE parameter
  of system disk • 3–4
Command procedures
  checking after upgrade • 9–3
Compact disc drive
  supported by UETP • 10–6
Computer-specific installation instructions • 3–1
Concurrent upgrade
  description • 7–1
  pre-upgrade procedure • 7–1 to 7–2
  procedure • 7–1 to 7–2
  setting QUORUM parameter • 7–2
  summary • 7–1
CONSCOPY.COM procedure • 6–4
Console
  connecting with SYSGEN • 9–8
Console media
  automatic update during upgrade • 6–4
  backing up after an upgrade • 9–13

Console media (Cont.)
backing up before an upgrade • 6–4
booting from during an upgrade • 8–6
copying • 6–4
modifying for automatic restart • 6–7
modifying to boot from [SYSF] • 8–6
required for an installation • 2–1
required for an upgrade • 5–4
restoring original boot command procedures • 8–12
updating after an upgrade • 9–7
using backup copy • 6–4
Creating a system disk • 3–3
Creating a transaction log file • 4–3, 9–3
CTRLNAME logical name • 10–8, 10–31
Customizing DECdtm services • 4–3, 9–3
Customizing the system • 4–2

D

Date
format for entering • 8–4
dddGEN
restoring original version after upgrade • 8–12
restoring original version during upgrade • 8–12
DECdtm services
customizing • 4–3, 9–3
disabling • 4–3, 9–3
DECnet
UETP defaults for installation • 10–28
DECnet node address • 2–3
providing during installation • 3–10
DECnet node name • 2–3
possible values • 3–10
providing during installation • 3–10
DECnet–VAX
configuration database • 9–4
configuring after an installation • 4–3
error message during UETP • 10–26
ignoring during startup • 4–3
preparing for UETP • 10–10
preparing for upgrade • 6–7
requirement for upgrade • 6–8
running with VMS DECwindows • 4–3
security measures • 9–4
UETP test of • 10–35
UETP test phase • 10–34
DECnet–VAX license
registering after installation • 4–2
registering during an upgrade • 8–14
DECnet–VAX license (Cont.)
registering during installation • 3–15
Decompressing system libraries • 4–4, 9–10
Decompressing the system messages help library • 4–5, 9–11
DECW$BOOK logical name • 11–2
DECW$IGNORE_DECNET logical name • 4–3, 11–1
DECW$IGNORE_DECWINDOWS logical name • 11–2
DECW$TAILOR
using to remove files from a system disk • 4–5, 9–12, 12–8, 12–9
DECwindows
See VMS DECwindows
DEFBOO
restoring original version after upgrade • 8–12
restoring original version during upgrade • 8–12
Device
format for upgrade • 8–3
HSC name format for upgrade • 8–3
Device names
determining • 2–4, 3–3
Devices
configuring before upgrade • 8–2
Device support files for VMS DECwindows • 8–9
Device test (in UETP)
running individually • 10–30
Diagnostics
relationship to UETP • 10–15
Digital Small Storage Interconnect (DSSI) disks
device name change • 3–2
Directories
user, processed during upgrade • 8–11
Directory trees
merged by upgrade procedure • 8–12
Disabling network objects • 9–4
Disabling the TP_SERVER process • 4–3, 9–3
Disk
See System disk, User disk
Disk drive
preparing for installation • 2–4
Disk space
amount needed for upgrade • 6–6
amount needed to decompress help library • 4–5, 9–11
amount needed to decompress libraries • 9–10
amount needed to decompress system libraries • 4–4
amount needed to run UETP • 4–4
increasing the amount on the system disk • 9–12
Distribution kit
magnetic tapes in • 3–3

Distribution kit (Cont.)
RL02 • 3–8
RX33 • 3–8
save sets in • 3–3
Standalone BACKUP • 3–2
TK50 tape cartridges in • 3–2
VMS • 3–2
VMS DECwindows • 3–2
with multiple volumes • 3–2, 8–11
Distribution media • 2–2
location during upgrade • 5–3
Dump file
checking size • 9–2
modifying size • 9–2
purged during upgrade • 8–10
requirement for upgrade • 6–5
shared • 5–3, 6–5

E

Editing startup files • 4–2
Enabling automatic restart • 6–7
Enabling manual reboot • 6–7
Error during installation
LICENSE • 3–5
Error during UETP • 10–19
diagnosing • 10–15
sources of • 10–16
Error during upgrade
cluster hang during shutdown • 7–4
during console update • 9–9
LIBRAR facility • 8–12
while entering device code • 8–3
Error formatter (ERRFMT)
stopped during upgrade • 8–6
Error Log Utility
relationship to UETP • 10–2, 10–15, 10–27
Error running ANALYZE/DISK_STRUCTURE • 6–5
Ethernet
defining a remote node for UETP • 10–17
preparing for UETP • 10–7
Example files for VMS DECwindows • 8–9
EXPECTED_VOTES
parameter • 9–4
setting for rolling upgrade • 7–4

F

FAL (File Access Listener) object
enabling or disabling • 9–5
Feedback data
checking before an upgrade • 6–2
FIELD account
setting password • 3–12
FILLM quota
checked during upgrade procedure • 8–6

H

Hang
See System hang
Help
setting up and decompressing • 4–4, 9–10
Hierarchical Storage Controller
See HSC (Hierarchical Storage Controller)
HLP$LIBRARY logical name • 4–4, 9–11
HSC (Hierarchical Storage Controller)
device format for an installation • 3–3
device format for upgrade • 8–3
displaying name • 2–5
drive • 2–2
if drive is connected to • 3–3
name • 3–3

I

Installation
See also Post-installation tasks
See also Post-installation tasks, Pre-installation tasks, Installation procedure
See also Pre-installation tasks
booting standalone BACKUP • 3–2
booting system disk after transferring required save set • 3–4
cautions and restrictions • 3–1
choosing optional VMS DECwindows components • 3–7
computer-specific instructions • 3–1
information needed for VAXcluster • 2–3
of layered products • 4–6
preparing for • 2–3
rebooting system • 3–16

Installation (Cont.)
recording • 2–3
registering licenses • 3–15
registering VMS license • 3–17
specifying VAXcluster configuration • 3–9
warning about VMSINSTAL option G • 3–2
Installation instructions • 3–1
Installation procedure • 1–1
choosing optional VMS components • 3–6
running AUTOGEN • 3–16
setting passwords • 3–12
stages of • 3–1
when to use • 1–1
Installing optional components • 3–5
Installing the VMS operating system • 3–1
Installing VMS DECwindows • 3–6
instructions • 3–8
Installing VMS on a VAXcluster • 3–9
Interrupt
upgrade before Phase 1 • 8–4

K

Keyboard layout • 11–3
customizing • 11–3
list of • 11–4
Keyboard model number • 11–4
Keymap name
determining • 11–4
KFQSA controller
upgrade requirement • 6–6

L

Layered products
effect of upgrade on • 5–3
installing after VMS installation • 4–6
supported versions • D–1
LIBDECOMP.COM (procedure to decompress system libraries) • 4–4
running after an upgrade • 9–10
Librarian Utility
See VMS Librarian Utility
Libraries
decompressing after an installation • 4–4
Library save set
contents • B–1
installing • 3–6
License Management Utility
using during installation • 3–15
Licenses
see also Product Authorization Key (PAK)
error messages • 3–5
registering after installation • 4–2
registering during an upgrade • 8–14
registering during VMS installation • 3–15
Line printer
preparing for UETP • 10–2, 10–4, 10–7
UETP output • 10–32
UETP test image • 10–33
UETP test of • 10–30
LOADS logical name • 10–33
Load test
defining user load for UETP • 10–13
description • 10–33
error during UETP • 10–25
running individually • 10–12
Local Area VAXcluster
information needed to install • 2–3
installation questions • 3–11
upgrade requirement • 5–3
Local drive • 2–2
Log
keeping of installation • 2–3
keeping of upgrade • 6–1
Log file
transaction
See Transaction log file
Log file generated by UETP
See also UETP.LOG
during the load test • 10–25
NETSERVER.LOG • 10–24
OLDUETP.LOG • 10–18
Logical names • 11–6
for VMS DECwindows • 11–2
Logical name used by UETP
CTRLNAME • 10–31
LOADS • 10–33
SYS$INPUT • 10–30
SYS$OUTPUT • 10–32
Long report format
See Console report during UETP

M

Magnetic tape
preparing for UETP • 10–2, 10–4, 10–5

Magnetic tape (Cont.)
test of • 10–30, 10–32
UETP test image • 10–33
Magnetic tape kit
description of contents • 3–3
MAIL object
enabling or disabling • 9–5
Mandatory update
applying during installation • 3–14
applying during upgrade • 8–13
Manual boot
after running AUTOGEN • 8–15
during upgrade procedure • 8–6, 8–12
enabling for upgrade • 6–7
Master command procedure
See UETP.COM
Messages
online help for • 4–4, 9–10
MicroVAX computers
booting from [SYS0] • 8–12
booting from [SYSF] • 8–10, A–1
Minimum startup
configuring devices after • 8–2
setting before upgrade • 6–6
MIRROR object
enabling or disabling • 9–6
Mixed-interconnect VAXcluster
information needed to install • 2–3
installation questions • 3–11
upgrade requirement • 5–3
Mixed-version cluster
upgrading to • 7–3 to 7–5
MODE • 10–38
MODE logical name • 10–16
MODPARAMS.DAT (parameter file) • 4–6
entering parameters in • 6–7
modifying for cluster • 9–4
modifying for single system • 9–4
Monochrome workstation • 11–3

N

NETCONFIG.COM procedure
running after an installation • 4–3
NETCONFIG_UPDATE.COM procedure
example • 9–5
running • 9–4
Network
See also DECnet–VAX
Network (Cont.)
requirement for upgrade • 6–8
restriction on upgrade • 5–2
Network objects
disabling • 9–4
Network security
updating after an upgrade • 9–4
NML (Network Management Listener) object
enabling or disabling • 9–6

O

Online help
for system messages • 4–4, 9–10
OPCOM
stopped during upgrade • 8–6
Operator logs
effect of upgrade on • 5–3
Optional components
selecting during installation procedure • 3–5
selecting during upgrade procedure • 8–7
Optional save set
contents • B–2
installing • 3–6
upgrading • 8–7
Output
saving from installation • 2–3
saving from upgrade • 6–1
Output during UETP
terminal and line printer • 10–32

P

Page file
checking size • 6–5, 9–2
modifying size • 6–5, 9–2
purged during upgrade • 8–10
required location • 6–5
size required for upgrade • 6–5
PAK
see Product Authorization Key (PAK)
Password
forgotten • 3–17, 8–16
generating for default DECnet account • 9–4
minimum length • 3–12
modifying during upgrade procedure • 8–5
requirements • 8–5
setting during installation • 3–12

Phase 1 of upgrade procedure • 8–5
Phase 2 of upgrade procedure • 8–10
Phase 3 of upgrade procedure • 8–11
Phase 4 of upgrade procedure • 8–12
Phase 5 of upgrade procedure • 8–13
Phase 6 of upgrade procedure • 8–13
Phase controller for UETP
  See UETPHAS00.EXE
PHONE object
  enabling or disabling • 9–6
Post-installation tasks • 4–1
  adjusting system parameters • 4–2
  backing up the system disk • 4–2, 4–6
  configuring DECnet–VAX • 4–3
  creating standalone BACKUP • 4–2
  customizing the system • 4–2
  customizing VMS DECwindows • 11–1
  decompressing system libraries • 4–4
  editing startup procedures • 4–2
  installing layered products • 4–6
  registering licenses • 4–2
  removing unwanted system files • 4–5
  setting up the system messages help library • 4–4
  setting up user accounts • 4–2
  testing the system with UETP • 4–4
  tuning the system with AUTOGEN • 4–6
Post-upgrade tasks
  adding and removing files • 9–12
  backing up console media • 9–13
  backing up system disk • 9–13
  changing MODPARAMS.DAT • 9–4
  checking system file sizes • 9–2
  creating standalone BACKUP • 9–13
  customizing or disabling DECdtm • 9–3
  customizing VMS DECwindows • 9–12, 11–1
  decompressing system libraries • 9–10
  examining command procedure templates • 9–3
  increasing space on the system disk • 9–12
  modifying size of dump file • 9–2
  modifying size of page file • 9–2
  modifying size of swap file • 9–2
  purging system files • 9–12
  replacing SYSUAF.DAT • 9–2
  running AUTOGEN • 9–13
  running NET_CONFIG_UPDATE.COM • 9–4
  running UETP • 9–11
  running UPDATE_CLUSTER_MEMBERS.COM • 9–7
  setting up the system messages help library • 9–10
  updating console media • 9–7
Post-upgrade tasks (Cont.)
  updating network configuration database • 9–4
  updating network security environment • 9–4
  using VMSTAILOR and DECW$TAILOR • 9–12
Pre-installation tasks • 2–3
  collecting information you will need • 2–5
  determining HSC name • 2–5
  preparing disk and tape drives • 2–4
  preparing for a VAXcluster environment • 2–3
Pre-upgrade tasks
  analyzing and repairing system disk • 6–5
  backing up console media • 6–4
  backing up system disk • 6–3
  checking page file size • 6–5
  configuring devices • 8–2
  creating standalone BACKUP • 6–4
  preparing DECnet–VAX • 6–7
  preparing system disk • 6–4
  preventing interactive logins • 6–7
  running AUTOGEN • 6–2
  setting system parameters • 6–6
  stopping queues • 6–8
Print Dialog Box
  logical names • 11–9
Printer
  recording installation on • 2–3
  recording upgrade on • 6–1
Print formats
  defining for VMS DECwindows • 11–9
Print queue
  associating with a print format • 11–9
Privilege
  required for UETP • 10–21
Product Authorization Key (PAK) • 3–15
  registering after an installation • 4–2
  registering during an installation • 3–15
  registering during installation • 3–15
  registering during upgrade • 8–14
Products (VMSINSTAL)
  specification for upgrade • 8–3
Programming Support for VMS DECwindows • 8–9
PURGE command • 9–12
Purging system files • 9–12

Q

Queues
  requirement for upgrade • 6–8
  stopping before upgrade • 6–8
Quorum

Quorum (Cont.)
See also VAXcluster quorum
maintaining during rolling upgrade • 7–4
Quorum disk
choosing during VAXcluster installation • 3–11
QUORUM parameter
setting for concurrent upgrade • 7–2
Quotas
checked during upgrade procedure • 8–6
required to run UETP • 10–21

R

Reboot
after installation • 3–16
manual • 6–7
Recording an installation • 2–3
Recording an upgrade • 6–1
Removing unwanted system files • 4–5
Required save set
transferring to system disk • 3–3
Restart
enabling automatic • 6–7
Restrictions
for VMS installation • 3–1
for VMS upgrade • 5–2
Rights database
creating during installation • 3–13
RIGHTSLIST.DAT
creating during installation • 3–13
RL02 distribution kit • 3–8, 8–11
RMS Journaling
registering during an upgrade • 8–14
RMS Journaling license
registering after installation • 4–2
registering during installation • 3–15
Rolling upgrade
compatibility problem with VMS version 5.3 • 7–5
description • 7–1
effect on SYSUAF.DAT • 6–5, 9–2
maintaining cluster quorum • 7–4
procedure • 7–3
source drive requirement • 7–4
VMS version requirement • 5–1
VMS version restrictions • 7–2
RV60 optical disk drive
supported by UETP • 10–6
RX33 distribution kit • 3–8, 8–11

S

Satellites nodes
upgrade requirement • 5–3
Security
enhancements to NETCONFIG.COM • 9–7
updating after an upgrade • 9–4
Selecting a disk or tape drive • 2–4
Selecting optional components • 8–7
Server
customizing the VMS DECwindows • 11–3
startup • 11–6
Setting passwords during an installation • 3–12
Shadowed system disk
upgrade restriction • 5–3
Short report format
See Console report during UETP
Shutdown
after installation • 3–16
before an upgrade • 6–4
Site-specific command procedures
restored during upgrade • 8–14, 9–3
Source-drive • 3–1
definition • 8–2
requirement for VAXcluster upgrade • 7–4
STABACKIT.COM (procedure to create standalone BACKUP) • 6–4
Standalone BACKUP
booting to begin an installation • 3–2
creating after an upgrade • 9–13
creating a kit • 4–2
creating before an upgrade • 6–4
on multiple volumes • 3–2
tape cartridge containing • 3–2
upgrade requirement • 6–4
Standalone BACKUP media
loading • 3–2
Startup files
editing after an installation • 4–2
STARTUP_P1 parameter
requirement for upgrade • 6–6
Swap file
checking size • 9–2
creating • 6–6
modifying size • 9–2
purged during upgrade • 8–10
required loction • 6–5
SWAPFILES.COM procedure
executing to change system file sizes • 9–2

SWAPFILES.COM procedure (Cont.)
to check or modify page file size • 6–5
to create swap file • 6–6
SYCONFIG.COM procedure
editing after an installation • 4–2
SYLOGICALS.COM procedure
editing after an installation • 4–2
SYLOGIN.COM procedure
editing after an installation • 4–2
SYS$DECDTM_INHIBIT logical name • 4–3, 9–3
SYS$INPUT logical name • 10–30
SYS$OUTPUT logical name • 10–32
SYS$TEST logical name • 10–3, 10–10, 10–18
[SYSF] directory
booting from during upgrade • 8–10, A–1
command to boot MicroVAX and VAXstation computers from • 8–10
[SYSF] directory tree
built by upgrade procedure • 8–7
deletion of • 8–13
SYSGEN
configuring devices • 8–2
parameter requirement for upgrade • 6–6
using to connect console • 9–8
SYSTARTUP_V5.COM
editing to control DECnet–VAX • 4–3
SYSTARTUP_V5.COM procedure
editing after an installation • 4–2
System
customizing • 4–2
logging into for UETP • 10–1, 10–3
resource requirements for UETP • 10–1, 10–3
SYSTEM account
forgotten password • 3–17, 8–16
setting password • 3–12
System disk • 2–2
analyzing and repairing • 6–5
backing up • 9–13
backing up after an installation • 4–6
backing up before an upgrade • 6–3
booting after installation • 3–16
booting after transferring required save set • 3–4
checking amount of free space on • 4–4
CLUSTER_SIZE parameter • 3–4
creating • 3–3
devices no longer supported • 3–1, 8–1
disk space needed to run UETP • 4–4, 10–4
free space needed for upgrade • 6–6
free space needed to decompress libraries • 4–4
preparing for a VMS installation • 2–5
preparing for a VMS upgrade • 6–4
System disk (Cont.)
removing unwanted files from • 4–5
required location during upgrade • 8–13
restrictions • 1–1
specifying volume label for • 3–5
test error during UETP • 10–22, 10–23
UETP test image • 10–33
UETP test of • 10–32
unique volume label requirement • 6–3
using DECW$TAILOR to customize • 4–5, 9–12, 12–8, 12–9
using VMSTAILOR to customize • 4–5, 9–12, 12–1
volume shadowing restriction • 5–3
System files
purging • 9–12
System Generation Utility (SYSGEN)
ALLOCLASS parameter • 2–4
System hang • 8–15, 10–19, 10–28
System messages
accessing with online help • 4–4, 9–10
System parameters
generated by AUTOGEN feedback • 6–2
requirement for upgrade • 6–6
SYSTEST account
logging into for UETP • 10–1, 10–3
privileges required for UETP • 10–21
quotas required to run UETP • 10–21
setting password • 3–12
SYSTEST directory
creating for UETP • 10–5
function during UETP • 10–3
SYSTEST_CLIG account
disabling during installation • 3–12
reenabling for UETP • 10–9
requirements for UETP • 10–9, 10–37
SYSUAF.DAT (User Authorization File)
effect of rolling upgrade on • 6–5, 9–2
replacing after upgrade • 9–2
required location • 6–5

T

Tailored system disk
See VMSTAILOR
Tape cartridge
containing standalone BACKUP • 3–2
containing VMS DECwindows save sets • 3–2
containing VMS save sets • 3–2
Tape cartridge drive

Tape cartridge drive (Cont.)
preparing for UETP • 10–6
Tape drive
preparing for installation • 2–4
Target disk
using backup copy for • 6–3
Target-drive • 3–1
definition • 8–2
TASK object
enabling or disabling • 9–5
TCP/IP software
use with DECwindows • 11–7
Terminal
preparing for UETP • 10–2, 10–4, 10–7, 10–13
simulating users for UETP • 10–33
test of • 10–30
UETP output • 10–32
UETP test image • 10–33
UETP test of • 10–32
Time
format for entering • 8–4
TLZ04 tape drive
supported by UETP • 10–6
TP_SERVER process
disabling • 4–3, 9–3
Transaction log file
creating • 4–3, 9–3
Transferring VMS save sets to system disk • 3–3
Troubleshooting
UETP • 10–19

U

UETCONT00.DAT file • 10–30
creation of • 10–30
UETINIDEV.DAT file • 10–29, 10–31
creation of • 10–30
format • 10–31
UETININET.DAT • 10–35
UETININET.DAT file • 10–35
UETINIT00.EXE image • 10–29
UETINIT01.EXE image • 10–19, 10–29
UETLOAD00.DAT • 10–33
UETNETS00.EXE • 10–35
UETP
See User Environment Test Package
UETP$NODE_ADDRESS logical name • 10–10
UETP (User Environment Test Package)
DECnet installation defaults • 10–28
RRD40 and RRD50 compact disc drive • 10–6
RV60 optical disk drive • 10–6
TLZ04 tape drive • 10–6
UETP.COM file • 10–29
termination of • 10–14
UETP.LOG file • 10–13, 10–18, 10–27, 10–34
UETPHAS00.EXE program • 10–29, 10–30
UETUNAS00.EXE UETP test image • 10–17
UIC
See User Identification Code
Unsupported system disks • 3–1, 8–1
Unsupported VAX computers • 3–1, 8–1
UPDATE_CLUSTER_MEMBERS.COM procedure
running • 9–7
UPDATE_CONSOLE.COM procedure
running after an upgrade • 9–10
Upgrade
See also Post-upgrade tasks, Pre-upgrade tasks, Upgrade procedure
automatic update of console media • 6–4
cautions and restrictions • 5–2
console media required • 5–4
disk space needed for page file • 6–5
disk space requirement • 6–6
effect on accounting logs • 5–3
effect on layered products • 5–3
effect on operator logs • 5–3
enabling automatic restart • 6–7
enabling manual reboot • 6–7
error entering device code • 8–3
files purged during • 8–10
interrupting before Phase 1 • 8–4
license requirements • 8–14
maintaining a secure enviroment • 5–1
modifying console media • 6–7
procedure • 8–2 to 8–16
recording • 6–1
required location of VMS save sets • 5–2
requirement for shared dump file • 6–5
requirements • 6–1
shared dump file • 5–3
using VMSINSTAL options G • 5–2
VMS version requirement • 5–1
Upgrade procedure • 1–3
applying mandatory update • 8–13
booting from console media • 8–6
booting from [SYSF] • 8–10
booting manually during • 8–6
building [SYSF] directory • 8–7
conversion of VAXVMSSYS.PAR • 8–13
correcting directory pointers • 8–12

Upgrade procedure (Cont.)
deletion of [SYSF] directory tree • 8–13
effect on command procedures • 9–3
effect on MODPARAMS.DAT • 9–4
files deleted by • 8–7, 8–11, 8–13
loading VMS DECwindows distribution media • 8–11
loading VMS distribution media • 8–2
manual reboot after • 8–15
manual reboot from [SYS0] • 8–12
merging directory trees • 8–12
modifying console media to boot from [SYSF] • 8–6
modifying passwords • 8–5
Phase 1 • 8–5
Phase 2 • 8–10
Phase 3 • 8–11
Phase 4 • 8–12
Phase 5 • 8–13
Phase 6 • 8–13
processing user files • 8–11
registering licenses • 8–14
removing VMS DECwindows distribution media • 8–11
removing VMS distribution media • 8–11
required location of system disk • 8–13
restoring original dddGEN • 8–12
restoring original DEFBOO • 8–12
restrictions • 1–3
running AUTOGEN • 8–15
security check • 8–13
selecting optional components • 8–7
selecting VMS DECwindows • 8–8
User Authorization File (UAF)
effect of rolling upgrade on • 6–5, 9–2
purged during upgrade • 8–10
replacing after upgrade • 9–2
required location • 6–5
User disk
preparing for UETP • 10–1, 10–5
space requirements for UETP • 10–4
test error during UETP • 10–22
UETP test image • 10–33
UETP test of • 10–32
User Environment Test Package (UETP)
aborting execution of • 10–14
description of • 10–1
disk space needed to run • 4–4
displaying tests as they run • 10–16
initialization phase • 10–29
interpreting output of • 10–16
master command procedure • 10–29
User Environment Test Package (UETP) (Cont.)
normal completion of • 10–14
organization of • 10–29
required privileges • 10–21
required quotas • 10–21
requirements for small disk systems • 10–10
running all phases of • 10–2
running individual phase of • 10–12
running multiple passes of • 10–12, 10–18
starting • 10–11
testing the VAX Vector Instruction Emulation Facility (VVIEF) • 10–11
testing vector processors • 10–11
typical failures reported by • 10–19
when to run • 4–4, 9–11
User files
processed during upgrade • 8–11
User Identification Code (UIC)
for UETP • 10–5
User load
defined for UETP DECnet–VAX test • 10–35
defining for the UETP load test • 10–13
equation used to determine for UETP load test • 10–17

V

VAX 11/730, 11/780, 11/785 computers
booting from [SYSF] • A–3
VAX 11/750 computers
booting from [SYSF] • A–2
VAX 6000 computers
booting from [SYSF] • A–5
VAX 8200, 8250, 8300, 8350 computers
booting from [SYSF] • A–2
VAX 8530, 8550 computers
booting from [SYSF] • A–1
VAX 8600, 8650 computers
booting from [SYSF] • A–3
VAX 8700 computers
booting from [SYSF] • A–1
VAX 8810 computers
booting from [SYSF] • A–1
VAX 8820, 8830, 8840 computers
booting from [SYSF] • A–1
VAX 8820-N computers
booting from [SYSF] • A–1
VAX 9000 computers
booting from [SYSF] • A–5
VAX C

VAX C (Cont.)
example programs • 11–2
extracting definition files • 11–2
VAXcluster
See also Cluster group number, Cluster password
building • 4–2
information required for VMS installation • 2–3
label for system disk • 6–3
preparing for UETP • 10–10
problem upgrading • 8–15
test failure during UETP • 10–24
types • 2–3
upgrade requirement for shared dump file • 6–5
VAXcluster installation
• 3–9
choosing quorum disk • 3–11
choosing VAXcluster cluster type • 3–10
CI-only VAXcluster • 3–10
configuring the cluster • 3–9
Local Area VAXcluster • 3–11
Mixed Interconnect VAXcluster • 3–11
suggested answers to questions • 3–9
supplying ALLOCLASS parameter • 3–11
VAXcluster license
registering after installation • 4–2
registering during installation • 3–15
registering during upgrade • 8–14
requirement • 3–10
VAXcluster quorum
maintaining during rolling upgrade • 7–4
maintaining during upgrade • 8–15
VAXcluster QUORUM parameter
setting for concurrent upgrade • 7–2
VAXcluster upgrade
See Concurrent upgrade
See Rolling upgrade
VAX computers
no longer supported • 3–1, 8–1
VAXstation computers
booting from [SYS0] • 8–12
booting from [SYSF] • 8–10, A–1
VAXVMSRL054
default volume label • 3–5
VAXVMSSYS.PAR parameter files
conversion • 8–13
Video font files for VMS DECwindows • 8–9
VMB.EXE
copying to console media • 9–7
VMS
optional components • 8–7
VMS DECwindows
applications • 8–9
automatic startup • 11–1
choosing options during an installation • 3–7, 8–8
compiling Ada software • 11–10
components • 3–8, 8–9
controlling startup • 11–1
customizing • 11–1
customizing after an installation • 4–6
customizing after an upgrade • 9–12
customizing the server • 11–3
defining print formats • 11–9
device support • 8–9
example C programs • 11–2
example files • 8–9
files • C–1
if you choose not to upgrade • 8–8
installing • 3–6, 3–8, 8–9
installing during the upgrade procedure • 8–8
installing selected portions • 8–8
installing the complete kit • 8–8
logical names • 11–2
optional components • 8–7
programming support • 8–9
selecting during upgrade procedure • 8–8
starting • 4–6
transports • 11–7
upgrading • 8–8
using customer-written transports • 11–7
video fonts • 8–9
VMS DECwindows distribution media
loading • 3–8, 8–11
removing from drive • 3–2, 3–9, 8–11
VMS DECwindows options
applications • 3–7
device support • 3–7
examples • 3–8
programming support • 3–8
video fonts • 3–7
VMS DECwindows save sets
location on magnetic tape • 3–3
tape cartridge containing • 3–2
VMS distribution media
loading • 3–3, 8–2
multiple volumes • 8–2
removing from drive • 3–8, 8–11
VMSINSTAL
invoke for upgrade • 8–2
option G • 3–2
VMS Installation
See Installation

VMS Librarian Utility
  using to set up online help • 4–5, 9–11
VMS library files
  selecting during upgrade • 8–7
VMS library save set
  contents • B–1
VMS license
  registering after installation • 4–2
  registering during VMS installation • 3–15
VMS optional files
  selecting during upgrade • 8–8
VMS optional save set
  contents • B–2
VMS save sets
  location on magnetic tape • 3–3
  required location • 3–2, 3–3
  tape cartridge containing • 3–2
  transferring to system disk • 3–3
VMSTAILOR
  using to install optional components • 3–6, 12–4
  using to remove files from a system disk • 4–5, 9–12, 12–3
VMS/ULTRIX Connection software (UCX)
  function • 11–7
VMS Upgrade
  See Upgrade
VMS upgrade and installation supplements
  using • 2–3, 3–1
VMS version required, for upgrade • 5–1
Volume label
  requirement for VAXclusters • 6–3
  specifying • 3–5
Volume Shadowing
  registering during an upgrade • 8–14
  upgrade restriction • 5–3
Volume Shadowing license
  registering after installation • 4–2
  registering during installation • 3–15
VOTES parameter
  checking for rolling upgrade • 7–3
  setting for concurrent upgrade • 7–2
  upgrade requirement • 7–2
VPM (VMS Performance Monitor) object
  enabling or disabling • 9–6

W

WELCOME.TXT
  updating • 9–3
Windowing system
  choosing • 3–12
WINDOW_SYSTEM SYSGEN parameter • 3–12
Workstation
  customizing • 11–3