VMS

digital

VMS Version 5.3 Upgrade and Installation Procedures

Order Number: AA–NG61B–TE

VMS Version 5.3 Upgrade and Installation Procedures

Order Number: AA–NG61B–TE

October 1989

This document describes the VMS Version 5.3 installation procedure, the upgrade procedure, and how to install VMS DECwindows software.

Revision/Update Information:	This manual supersedes the *VMS Version 5.2 Upgrade and Installation Procedures*, Version 5.2.
Software Version:	VMS Version 5.3

digital equipment corporation
maynard, massachusetts

October 1989

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 1989.

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

The following is a third-party trademark:

PostScript is a registered trademark of Adobe Systems, Inc.

ZK5314

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 xiii

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 1–2

PART I

CHAPTER 2 INSTALLING THE VMS OPERATING SYSTEM AND DECWINDOWS SOFTWARE 2–1

2.1 BEFORE YOU START 2–1

2.2 TURNING ON THE SYSTEM 2–2

2.3 PREPARING THE DISK AND TAPE DRIVES 2–3

2.4 BOOTING STANDALONE BACKUP ON A VAX 8800 SERIES COMPUTER 2–4

2.5 BOOTING STANDALONE BACKUP 2–4

2.6 CREATING THE SYSTEM DISK 2–5

2.7 INSTALLING VMS DECWINDOWS SOFTWARE 2–10

2.8 JOINING A VAXCLUSTER 2–10

2.9 SETTING PASSWORDS AND RUNNING AUTOGEN 2–13

CHAPTER 3 AFTER INSTALLING VMS 3–1

3.1 REGISTERING YOUR LICENSES 3–1

3.2 STARTING VMS DECWINDOWS SOFTWARE 3–2

3.3 CUSTOMIZING THE SYSTEM 3–2

3.4 TESTING THE SYSTEM 3–2

3.5 DECOMPRESSING THE SYSTEM LIBRARIES 3–2

3.6 REMOVING UNWANTED FILES 3–3
3.6.1 Using VMSTAILOR 3–3
3.6.2 Using DECW$TAILOR 3–4

3.7 BACKING UP THE SYSTEM DISK 3–5

3.8 RUNNING AUTOGEN 3–5

PART II

CHAPTER 4 OVERVIEW OF THE UPGRADE PROCEDURE 4–1

4.1 WHAT HAPPENS DURING AN UPGRADE 4–2

4.2 CAUTIONS AND RESTRICTIONS 4–2

4.3 WHAT YOU NEED FOR THE UPGRADE 4–3

CHAPTER 5 BEFORE YOU DO AN UPGRADE 5–1

5.1 UNDERSTANDING VERSION 5.3 OF THE VMS OPERATING SYSTEM 5–1

5.2 AUTOGEN FEEDBACK PARAMETERS 5–1

5.3 BACKUP PROCEDURES 5–2

5.4 PREPARING THE SYSTEM DISK 5–3

5.5 PREPARING THE SYSTEM 5–5

5.6 SUMMARY 5–6

CHAPTER 6 UPGRADING A VAXCLUSTER ENVIRONMENT 6–1

6.1 PERFORMING A CONCURRENT UPGRADE 6–1

6.2 ROLLING UPGRADE OVERVIEW 6–2

6.2.1 Performing a Rolling Upgrade 6–2

CHAPTER 7 STANDALONE UPGRADE 7–1

7.1 BEGINNING THE UPGRADE PROCEDURE 7–1

7.2 UPGRADE PHASE 1 7–4

7.3	UPGRADE PHASE 2	7–8
7.4	UPGRADE PHASE 3	7–9
7.5	UPGRADE PHASE 4	7–10
7.6	UPGRADE PHASE 5	7–10
7.7	UPGRADE PHASE 6	7–11

CHAPTER 8 AFTER YOU DO AN UPGRADE 8–1

8.1	REPLACING SYSUAF.DAT	8–1
8.2	EXAMINING YOUR COMMAND PROCEDURES	8–1
8.3	CHANGING MODPARAMS.DAT	8–2
8.4	RUNNING DECNET–VAX	8–2
8.4.1	Running NETCONFIG_UPDATE.COM	8–2
8.4.2	Running UPDATE_CLUSTER_MEMBERS.COM	8–4
8.5	UPDATING YOUR CONSOLE MEDIA	8–5
8.5.1	VAX 8530, 8550, 8700, 8800, 8810, 8820-N	8–5
8.5.2	VAX 11/730, 750, 780, 785 and VAX 8200, 8250, 8300, 8350, 8600, 8650	8–7
8.6	DECOMPRESSING THE SYSTEM LIBRARIES	8–8
8.7	TESTING THE SYSTEM WITH UETP	8–8
8.8	INCREASING FREE SPACE ON THE SYSTEM DISK	8–8
8.8.1	Using VMSTAILOR	8–8
8.8.2	Using DECW$TAILOR	8–9
8.8.3	Checking the Size of System Files	8–10

8.9 STARTING AND CUSTOMIZING VMS DECWINDOWS SOFTWARE 8–11

8.10 BACKING UP THE SYSTEM DISK 8–11

8.11 RUNNING AUTOGEN 8–11

CHAPTER 9 RUNNING UETP 9–1

9.1 SUMMARY OF UETP OPERATING INSTRUCTIONS 9–1

9.2 LOGGING IN 9–3
9.2.1 SYSTEST Directories 9–3

9.3 SETTING UP FOR UETP 9–3
9.3.1 Setting Up the System Disk 9–4
9.3.2 Setting Up Additional Disks 9–4
9.3.3 Setting Up Magnetic Tape Drives 9–5
9.3.4 Setting Up Tape Cartridge Drives 9–6
9.3.5 Setting Up Terminals and Line Printers 9–6
9.3.6 Preparing Ethernet Adapters for UETP Testing 9–6
9.3.7 Preparing the DR11–W for UETP Testing 9–6
9.3.8 Preparing the DPV11–WA for UETP Testing 9–7
9.3.9 Preparing the DR750 or the DR780 for UETP Testing 9–7
9.3.10 Preparing the MA780 for UETP Testing 9–8
9.3.11 Preparing a Second LPA11–K for UETP Testing 9–8
9.3.12 Devices Not Tested 9–8
9.3.13 Preparing for VAXcluster Testing 9–8
9.3.14 Preparing a Small-Disk System 9–9
9.3.15 Preparing DECnet–VAX 9–10

9.4 STARTING UETP 9–10
9.4.1 Running a Subset of Phases 9–11
9.4.2 Single Run Versus Multiple Passes 9–11
9.4.3 Defining User Load for Load Test 9–12
9.4.4 Long and Short Report Format 9–12
9.4.5 Termination of UETP 9–13
9.4.5.1 Using CTRL/Y • 9–13
9.4.5.2 Using CTRL/C • 9–14

9.5	**TROUBLESHOOTING**	**9–14**
9.5.1	**Relationship of UETP to Error Logging and Diagnostics**	**9–14**
9.5.2	**Interpreting UETP Output**	**9–14**
9.5.2.1	Defining a Remote Node for UETP Ethernet Testing • **9–16**	
9.5.3	**The Log Files**	**9–17**
9.5.4	**Possible UETP Errors**	**9–18**
9.5.4.1	Wrong Quotas, Privileges, or Account • **9–19**	
9.5.4.2	UETINIT01 Failure • **9–20**	
9.5.4.3	Insufficient Disk Space • **9–22**	
9.5.4.4	Incorrect Setup of a VAXcluster • **9–23**	
9.5.4.5	Problems During the Load Test • **9–24**	
9.5.4.6	DECnet–VAX Error • **9–25**	
9.5.4.7	Errors Logged but Not Displayed • **9–25**	
9.5.4.8	No PCB or Swap Slots • **9–26**	
9.5.4.9	Hangs • **9–26**	
9.5.4.10	Bugchecks and Machine Checks • **9–27**	

9.6	**UETP TESTS AND PHASES**	**9–27**
9.6.1	**Initialization Phase**	**9–28**
9.6.2	**Device Test Phase**	**9–28**
9.6.2.1	How the Device Phase Works • **9–28**	
9.6.2.2	Running a Single Device Test • **9–29**	
9.6.3	**System Load Test Phase**	**9–31**
9.6.4	**DECnet–VAX Test Phase**	**9–32**
9.6.4.1	Environment • **9–32**	
9.6.4.2	How the DECnet–VAX Phase Works • **9–33**	
9.6.5	**Cluster-Integration Test Phase**	**9–35**

CHAPTER 10 STARTING AND CUSTOMIZING VMS DECWINDOWS SOFTWARE **10–1**

10.1	**STARTING THE VMS DECWINDOWS SOFTWARE**	**10–1**
10.1.1	**After an Installation**	**10–1**
10.1.2	**After an Upgrade**	**10–2**
10.1.3	**VMS DECwindows Logical Names Created At Startup**	**10–2**

10.2	**CUSTOMIZING THE SERVER STARTUP**	**10–3**

10.3	USING TCP/IP WITH DECWINDOWS	10–6
10.4	USING CUSTOMER-WRITTEN TRANSPORTS WITH DECWINDOWS	10–7
10.5	USING THE EXAMPLE TRANSPORT	10–7
10.6	DEFINING PRINT FORMATS	10–8
10.7	COMPILING ADA SOURCE FILES	10–9

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

A.1	MICROVAX AND VAXSTATION COMPUTERS	A–1
A.2	VAX 8530, 8550, 8700, 8810-N, 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	VAXSTATION 8000	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

APPENDIX E RELEASE NOTES FOR THE VAX 8200 E–1

E.1 HOW TO PROCEED E–1

E.2 INSTALLING THE VMS OPERATING SYSTEM E–1

E.3 EDITING CIBOO.CMD E–3

APPENDIX F USING DECW$TAILOR TO INSTALL DECWINDOWS F–1

INDEX

TABLES

2–1 Installation Questions for CI-Only Configurations 2–11

2–2 Installation Questions for Local Area and Mixed-Interconnect Configurations 2–12

9–1 The Device Tests 9–31

10–1 DECW$LOGICAL_NAMES 10–2

10–2 DECwindows Keymap 10–4

10–3 DECW$SERVER(n)_TABLE 10–6

10–4 Print Dialog Box Logical Names 10–9

C–1 DECwindows Software Components C–1

D–1 Layered Products Currently Available for VMS Version 5.3 D–1

Preface

This manual contains information about the following:

- The VMS Version 5.3 installation procedure
- The VMS Version 5.3 upgrade procedure
- Installing VMS DECwindows software

If you are going to install the VMS operating system, follow the installation procedure in this guide. Refer to the VMS installation and operations guide for your VAX computer for information specific to your computer. Information specific to your computer includes such tasks as turning on the system, booting standalone BACKUP, and booting the new system disk.

If you are going to upgrade to Version 5.3 of the VMS operating system, follow the upgrade procedure in this guide.

Intended Audience

This manual is intended for system managers, operators, and users of the VMS operating system. Read this manual before you install, upgrade, or use Version 5.3 of the VMS operating system.

Document Structure

This manual contains the following chapters:

- Chapter 1 describes what happens during an installation or an upgrade.
- Chapter 2 describes how to install Version 5.3 of the VMS operating system.
- Chapter 3 lists the tasks you should perform after you install the VMS operating system.
- Chapter 4 contains cautions, restrictions, notes, and suggestions about upgrading your VAX computer system.
- Chapter 5 tells you how to prepare your system for an upgrade.
- Chapter 6 describes the procedures for upgrading a VAXcluster environment.
- Chapter 7 describes the procedures for upgrading a single computer system.
- Chapter 8 lists the tasks you should perform after the upgrade.
- Chapter 9 describes the VMS User Environment Test Package (UETP) and how to use it to test the system.
- Chapter 10 describes how to start and customize VMS DECwindows software.

- 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 5.3 of the VMS operating system.
- Appendix E contains release notes for the VAX 8200 computer.
- Appendix F contains information on how to install VMS DECwindows sofware using DECW$TAILOR (after an initial VMS installation).

Associated Documents

The following documents may be useful:

- The installation and operations guide for your VAX computer–Provides information that you will need during an installation or upgrade, such as booting and device naming procedures.
- *VMS Version 5.3 New Features Manual*–Describes new software features of VMS Version 5.3.
- *VMS VAXcluster Manual*–Describes the procedures for setting up and managing VAXcluster configurations.
- *VMS Version 5.3 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.
. . .	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.
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.

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 5.3 of the VMS operating system on the following VAX computers and system disks:

- **MicroVAX I, VAXstation I, and VAX 11/725**
- **RK07, RL02, RC25, 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.

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

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.**

If you are upgrading your system, go to Chapter 4.

Part I

Part I contains information about installing Version 5.3 of the VMS operating system.

2 Installing the VMS Operating System and DECwindows Software

This chapter describes installing Version 5.3 of the VMS operating system and VMS DECwindows software. Use this chapter in conjunction with the VMS installation and operations guide for your VAX computer. The VMS installation and operations guide has information specific to your computer, such as turning on the system, booting standalone BACKUP, and booting the new system disk.

CAUTION: You cannot install or upgrade to Version 5.3 of the VMS operating system on the following VAX computers and system disks:

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

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.

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 4.

2.1 Before You Start

Before you install the VMS operating system, do the following:

- 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.
- 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.

CAUTION: Do not attempt to use VMSINSTAL with OPTIONS 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.

- The VMS save sets must be in the [000000] directory on the system disk. The save sets are in this directory on all Digital-supplied distribution kits. If you have to move the VMS save sets, they must be in the [000000] directory, or the installation will fail.

- 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.
- Read the "Before Installing VMS" chapter in the VMS installation and operations guide for your VAX computer before you do the installation. This chapter contains important information, specific to your VAX computer.
- 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 configuration. For a complete description of configurations, see the *VMS VAXcluster Manual*. Depending on the type of configuration, you need to obtain the following information from either the network or VAXcluster manager:
  - — CI-Only Configuration: Obtain the allocation class value, the DECnet node name, and DECnet node address for the computer.
  - — Local Area and Mixed-Interconnect Configurations: Obtain the allocation class value, the DECnet node name, and DECnet node address for the computer. You also need the cluster group number and password.
- If you have a VAX computer that uses console media, you also need the most recent version of the console media. For example, if you have a VAX–11/780, you need the most recent version of the console RX01.

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 is the list of tasks in the installation procedure:

1 Turning on the system

2 Preparing the disk and tape drives

3 Booting standalone BACKUP

4 Creating the system disk

5 Installing VMS DECwindows software (optional)

6 Joining a VAXcluster (optional)

7 Setting passwords and running AUTOGEN

To install the VMS operating system, follow the directions in this chapter. At certain points you will need to refer to the VMS installation and operations guide for your VAX computer.

2.2 Turning On the System

If necessary, read the section "Turning on the System" in the VMS installation and operations guide for your VAX computer. Once the system is turned on, go to Section 2.3 in this chapter to prepare the disk and tape drives.

2.3 Preparing the Disk and Tape Drives

Set up the disk and tape drives you use during the installation by doing the following:

1 Decide which drive will hold the distribution tape or disk and which drive will hold the system disk. Determine the device names for each drive. Write these names on a piece of paper. You will need this information throughout the installation. If you need more information about determining the device names for your system, refer to the VMS installation and operations guide for your VAX computer.

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 Hierarchical Storage Controller (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.

5 To boot standalone BACKUP on a VAX 8800 series computer, go to Section 2.4.

To boot standalone BACKUP on any other VAX computer, go to Section 2.5.

2.4 Booting Standalone BACKUP On a VAX 8800 Series Computer

Follow the instructions in this section only if you are booting standalone BACKUP on a VAX 8800 series computer. If you are booting standalone BACKUP on any other VAX computer, go to Section 2.5.

To boot standalone BACKUP on a VAX 8800 series computer, do the following:

1 Use the TK50 tape cartridge labeled VMS V5.2 CONSOLE S/A BKUP TK50 DEC to copy standalone BACKUP to the hard disk of your your console subsystem before beginning the software installation.

- If you are a new VAX 8800 Series customer, you will find this TK50 tape cartridge in your software kit.
- If you are a service update customer, you received this TK50 tape cartridge with your VMS Version 5.2 software kit.

VMS Installation and Operations: VAX 8820, 8830, 8840 contains instructions for copying standalone BACKUP to the hard disk of your console subsystem. Digital is not providing a new standalone BACKUP kit for VAX 8800 series computers with VMS Version 5.3 because this kit has not changed since VMS Version 5.2.

2 Boot standalone BACKUP from the hard disk of your console subsystem, following the instructions in the "Booting Standalone BACKUP" section of the *VMS Installation and Operations: VAX 8820, 8830, 8840*. After you boot standalone BACKUP, return to this guide for the next step in the VMS Version 5.3 installation procedure.

3 After you boot standalone BACKUP you should see the following display:

```
%BACKUP-I-IDENT, standalone BACKUP V5.3; the date is 19-APR-1990 15:00
$
```

4 Go to Section 2.6 in this chapter to create the system disk.

2.5 Booting Standalone BACKUP

Note: Read this note only if your system contains Digital Small Storage Interconnect (DSSI) disks attached to a KFQSA controller. Starting with VMS Version 5.3, you will notice a change in the device names for DSSI disks attached to KFQSA controllers. Digital implemented this naming change so that the VMS operating system could support additional configuration types. For more information on supported configurations, see the *VMS Version 5.3 Release Notes*.

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 disk attached to KFQSA controllers, see the *VMS Version 5.3 Release Notes*.

To boot standalone BACKUP, complete the following steps:

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 V5.3 BIN TK50 2/2 DECWINDOWS & S/A BKUP*. To start the installation, put this tape cartridge in the appropriate drive.

2. Follow the instructions in "Booting Standalone BACKUP" in the VMS installation and operations guide for your VAX computer. After you boot standalone BACKUP, return to this guide for the next step in the VMS Version 5.3 installation procedure.

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

```
%BACKUP-I-IDENT, standalone BACKUP V5.3; the date is 19-APR-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 V5.3 BIN TK50 1/2 VMS BINARY*.

4. Go to Section 2.6 in this chapter to create the system disk.

2.6 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 V5.3 magnetic tape kit contains VMS and VMS DECwindows software on two tapes. The first tape contains the VMS *required* and *library* save sets. The second tape contains the *optional* save set and 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.

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

[1] If the distribution media is a disk, the VMS save sets are in the [0,0] directory. They must be in this directory for the installation procedure to work.

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 distribution tape cartridge, make sure the tape labeled *VMS V5.3 BIN TK50 1/2 VMS BINARY* is in the *source-drive*.

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

```
$ BACKUP/VERIFY [hsc-name]source-drive:VMS053.B/SAVE_SET [hsc-name]target-drive:
```

Substitute the appropriate device names for *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 VMS053. 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 MSA0.
- A *target-drive* that is an RA60 disk drive with a device name of DJA1.

Enter the following command and press RETURN:

```
$ BACKUP/VERIFY MSA0:VMS053.B/SAVE_SET DJA1:
```

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/VERIFY MUC6:VMS053.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 system 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. When the standalone BACKUP is done checking the files, the procedure displays the following message:

[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 installation and operations guide for your VAX computer.

```
%BACKUP-I-PROCDONE, Operation completed. Processing finished at 19-APR-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 installation and operations guide for your VAX computer. After you boot the new system disk, return to this guide for the next step in the VMS Version 5.3 installation procedure.

6 When the boot is complete, the procedure displays a message and asks you for the date and time. Enter the date and time using the 24-hour clock format and press RETURN. For example:

```
  VAX/VMS Version BI53-xxx Major version id = 1 Minor version id = 0

     VAX/VMS Version V5.3 Installation Procedure

                              Model: VAX 8800
                      System device: RA60
                        Free blocks: XXXXXX

* Please enter the date and time (DD-MMM-YYYY HH:MM): 19-APR-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-APR-1990 15:00:00.00  %%%%%%%%%%
     .
     .
     .
```

Note: The procedure displays warning messages that the VMS license is not registered. Continue with the installation procedure and register the VMS license after you finish the installation. For more information about registering your licenses, see the *VMS License Management Utility Manual.*

7 The procedure asks for 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, VAXVMSRL053, 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 VAXVMSRL053 just press RETURN in response to
the next question.

* Enter the volume label for this system disk [VAXVMSRL053]: SYSDSK
```

8 The procedure asks 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, enter the following 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

    The full VMS kit will be installed on this system disk.
    The following (with approximate sizes) will be provided:

        -VMS library                    - 27500 blocks
        -VMS optional                   -  9000 blocks

     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)
```

If you *do not* want to install VMS DECwindows software, type N (for NO), press RETURN, and go to step 9. 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.

NOTE: If you do not install VMS DECwindows at this time, you can install it after the installation using DECW$TAILOR. Please note that for TK50 tape cartridges, it is quicker to install VMS DECwindows software during the installation than to tailor VMS DECwindows after the upgrade. For more information about using DECW$TAILOR to install VMS DECwindows software, see Appendix F.

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

a. Type Y (for YES) 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 44500 blocks. If you
want to perform a partial installation, you can choose to install one
or more of the following DECwindows components:

-DECwindows device support      -   9500 blocks
-DECwindows applications        -  11000 blocks
-DECwindows programming support -  24000 blocks

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

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

To install selected portions of the VMS DECwindows kit, type N (for NO) 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 the applications files.

- Applications files—VMS DECwindows provides a variety of applications including the following: Bookreader, Calculator, Calendar, Cardfiler, Clock, DDIF Document Viewer, 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 (100DPI) 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.
- 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.

c. 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 answer NO to this prompt, you can respecify the VMS DECwindows options that you want to install.

9 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
```

The contents of the VMS library and optional save sets is listed in Appendix B. After you finish the installation, you can use VMSTAILOR to remove the VMS files you do not need. For more information on VMSTAILOR, see Chapter 3.

10 If you are installing VMS DECwindows software, go to Section 2.7.

If you are not installing VMS DECwindows software, go to Section 2.8.

2.7 Installing VMS DECwindows Software

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

1 Remove the media 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:.

Put the DECwindows distribution kit in the drive.
```

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

Note: If you have either an RL02 or RK07 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 Enter 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, DECW53 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
```

The VMS DECwindows files are listed in Appendix C. After you finish the installation, you can use DECW$TAILOR to remove the VMS DECwindows files you do not need. For more information on removing files with DECW$TAILOR, see Chapter 3.

5 To continue the installation procedure, go to Section 2.8.

2.8 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 the VMS operating system in a VAXcluster environment, the procedure then asks a series of questions about your configuration.

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 the VMS operating system on a standalone system, type N (for NO), press RETURN.

If you are installing the VMS operating system in a VAXcluster environment, type Y (for YES) and press RETURN. If you answer YES to the VAXcluster question, you must have a VAXcluster license.

2 If you installed VMS DECwindows, the procedure displays the following message:

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

Depending on your preference, type Y (for YES) or N (for NO) and press RETURN. If you type Y (for YES), the procedure sets the SYSGEN WINDOW_SYSTEM parameter to one for DECwindows. If you type N (for NO), the procedure sets the SYSGEN WINDOW_SYSTEM parameter to zero.

3 If you are installing the VMS operating system in a VAXcluster environment, go to step 4.

If you are installing the VMS operating system on a standalone system, go to step 6.

4 The procedure displays the following message:

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

5 Determine the type of cluster configuration you want to create (configuration types are described in the *VMS VAXcluster Manual*). Table 2–1 lists the questions you are asked if you want a CI-only configuration. Table 2–2 lists the questions you are asked if you want a local area VAXcluster or a mixed-interconnect configuration. Typical responses are explained in the tables.

Table 2–1 Installation Questions for CI-Only Configurations

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 N. The Ethernet is not used for cluster (SCS internode) communications in CI-only configurations.

(continued on next page)

Table 2–1 (Cont.) Installation Questions for CI-Only Configurations

Question	How to Respond
Will KRUP be a disk server (Y/N)?	Enter Y or N, depending on your configuration requirements. 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.
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 2–2 Installation Questions for Local Area and Mixed-Interconnect Configurations

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 configurations.
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.
Will KRUP be a disk server (Y/N)?	Enter Y. In local area and mixed-interconnect configurations, the system disk is always served to the cluster. Refer to the *VMS VAXcluster Manual* for information on served cluster disks.

(continued on next page)

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

Question	How to Respond
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 configuration, 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.

6 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 and go to Section 2.9.

7 Continue with Section 2.9.

2.9 Setting Passwords and Running AUTOGEN

Follow the directions in this section to set passwords and run AUTOGEN. AUTOGEN evaluates your hardware configuration and estimates typical workloads. It then sets system parameters, the sizes of the page, swap, and dump files, and the contents of VMSIMAGES.DAT. When AUTOGEN finishes and you reboot, the installation procedure is complete.

1 The procedure asks you for new passwords for the SYSTEM, SYSTEST, and FIELD accounts. Passwords must be at least eight 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 (for NO) and press RETURN. When the procedure asks if you want to enter a new password, type Y (for YES) 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 there will be a
negative impact on the performance of 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 installation and operations
guide 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

============================================================
Continuing with VAX/VMS V5.3 installation procedure.

Configuring all devices on the system.

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

```
After the system has rebooted you must register any Product
Authorization Keys (PAKs) that you have received with this kit.
You may register these PAKs by executing the following procedure:

    $ @SYS$UPDATE:VMSLICENSE

See the VMS License Management Utility Manual for any additional
information you may need.

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

4 AUTOGEN runs and 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.
      .
      .
      .
```

5 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
```

6 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 information on booting, refer to the VMS installation and operations guide for your VAX computer. Then return to this guide for the next step in the VMS Version 5.3 installation procedure.

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
```

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

```
   VAX/VMS Version V5.3 Major version id = 01 Minor version id = 00

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

VAX/VMS V5.3

You have SUCCESSFULLY installed the VMS operating system.

The system is now executing the STARTUP procedure. Please
await the completion of STARTUP before logging into the
system (approximately three minutes).

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

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

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

Note: The procedure displays warning messages that the VMS license must be registered. Be sure to register this license when the installation procedure finishes, as described in the *VMS License Management Utility Manual.*

Finally, the procedure displays informational messages as well as 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-APR-1990 15:00:00.00
  SYSTEM       job terminated at 19-APR-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.

8 Press RETURN. 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 V5.3

Username: SYSTEM
Password:
     .
     .
     .
     Welcome to VAX/VMS Version V5.3
```

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

9 There are several things you must do before you can use the system. For complete information, see Chapter 3.

3 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 tells you what the tasks are, whether they are optional or required, and the order in which you perform them. The following list summarizes the tasks that are described in this chapter:

1 Registering your licenses—You must register the VMS license that came with the software. You must also register the licenses you have for any system integrated products that you purchased.

2 Customizing and starting VMS DECwindows—After you register the VMS license you can customize and start the VMS DECwindows environment.

3 Customizing the system—Depending on whether you have a standalone system or a system that is part of a VAXcluster environment, there are several things you must do.

4 Testing the system—Once you have customized the system, run the VMS User Environment Test Package (UETP) to test the system.

5 Decompressing the system libraries—After you test the system, you can decompress the system libraries.

6 Removing unwanted files from the system disk —You can free up space on the system disk by removing the VMS and DECwindows files that you do not need.

7 Backing up the system disk—To protect all the work you have just done, make a backup copy of the system disk.

8 Running AUTOGEN—AUTOGEN fine tunes the system parameters for you.

3.1 Registering Your Licenses

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

After you register the VMS license, you must 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.*

3.2 Starting VMS DECwindows Software

If you have installed VMS DECwindows software, there are several steps that you must take to start and customize the VMS DECwindows environment. Chapter 10 describes these steps. After you finish starting and customizing VMS DECwindows software, return to this chapter.

For more information on starting and configuring DECnet–VAX, see the *Guide to DECnet–VAX Networking*.

3.3 Customizing the System

You must customize the system disk so that it automatically performs certain tasks when you boot. In addition, if your VAX computer is part of a VAXcluster environment, you must prepare the cluster operating 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 installation and operations guide 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. If you have a standalone system, 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

3.4 Testing the System

Run the User Environment Test Package (UETP) to verify the installation. For complete information, see Chapter 9. Note that UETP needs at least 1200 free blocks on the system disk.

3.5 Decompressing the System Libraries

Decompressing the system libraries gives the system faster access to the libraries. The decompressed libraries require approximately 8000 additional blocks of disk space. To find out how much disk space you have, enter the following command and press RETURN:

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

If you have at least 8000 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
```

3.6 Removing Unwanted Files

Read this section if you want to remove some of the VMS operating system and DECwindows files that you do not need from the system disk.

3.6.1 Using VMSTAILOR

To remove unwanted VMS files from the system disk, use VMSTAILOR. For example, if you are not running DECnet–VAX, you do not need the network support files. To remove unwanted files with the VMSTAILOR program, do the following:

1 Log in to the SYSTEM account.

2 Enter the following command, and press RETURN:

```
$ RUN SYS$UPDATE:VMSTAILOR
```

VMSTAILOR displays the following message:

```
.
.
.
Do you want to tailor files "ON" or "OFF"?
```

3 Type OFF to remove unwanted files.

The VMSTAILOR program lists each group 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 group. A file group is made up of many small subgroups. You can eliminate an entire group of files, or you can eliminate one or more of its subgroups. For example:

```
CLASS - Network support
        Size of entire class (with subclasses): 1478
        Size of common files required for any subclass: 1233
        Do you wish to select the entire class (default = NO)?
```

4 Decide which file groups or subgroups you do not need to support your system. The VMSTAILOR program displays step-by-step instructions.

5 VMSTAILOR displays the names of the files it deletes. After it finishes, AUTOGEN runs automatically to make the adjustments that are necessary after system files are deleted. AUTOGEN also reboots the system.

Note: You can use VMSTAILOR at any time to delete or add groups of VMS files to the system disk. 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 5.3 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 5.3 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:VMSUn053.TXT.

3.6.2 Using DECW$TAILOR

To remove unwanted DECwindows files from the system disk, use DECW$TAILOR. For example, if you are not planning on utilizing the DECwindows programming support, you can remove those files. To remove DECwindows files from the system disk, do the following:

1 Log in to the SYSTEM account.

2 Enter the following command, and press RETURN:

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

DECW$TAILOR displays the following message:

```
.
.
.
Do you want to tailor files "ON" or "OFF"?
```

3 Type OFF to remove unwanted files.

The DECW$TAILOR program lists each group of files and its size in blocks. Files are grouped according to their function. For example, all the files required for VMS DECwindows user support are in one group. A file group is made up of many small subgroups. You can eliminate an entire group of files, or you can eliminate one or more of its subgroups. For example:

```
CLASS - DECwindows user support files
          Size of entire class (with subclasses): 8456
          No subclasses in this tailor class.
          Do you wish to select the entire class (default = NO)?
```

4 Decide which file groups or subgroups you do not need to support your system. The DECW$TAILOR program displays step-by-step instructions.

When tailoring VMS DECwindows files, consider the following:

- Applications files—VMS DECwindows provides a variety of applications; including the following: Bookreader, Calculator, Calendar, Cardfiler, Clock, DDIF Document Viewer, 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. You may also want these files on a nonworkstation VAX computer, because you can run VMS DECwindows applications

on a nonworkstation VAX computer and display them on a workstation.

- 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.

Note: If you tailor device support files ON or OFF, the system reboots.

- 100 dots-per-inch (100DPI) 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.
- 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.

5 DECW$TAILOR displays the names of the files it deletes. If you deleted device support files, AUTOGEN runs automatically and reboots the system.

3.7 Backing Up the 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. Digital recommends that you perform the following operations:

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

For complete information on these operations, see the installation and operations guide for your VAX computer. Once you have backed up the system disk, install any software products that you have purchased. Follow the directions given in the software product manuals.

3.8 Running AUTOGEN

Run AUTOGEN FEEDBACK and reboot the system after 24 hours of operation, and again two 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$FEEDBACK.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 II

Part II describes how to upgrade to Version 5.3 of the VMS operating system.

4 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 5.3 upgrade procedure gives you the opportunity to install VMS DECwindows software on the system disk.

Note: The Version 5.3 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.

The Version 5.3 upgrade procedure leaves your VMS license intact. You do not need to re-install the VMS license after you upgrade. (See the *VMS License Management Utility Manual* for detailed information on license management for VMS Version 5.3. For example, starting with VMS Version 5.2 you must register your VAXcluster license before you can use your VAXcluster software.)

Before you can upgrade to VMS Version 5.2, your system must be running at least Version 5.0 of the VMS operating system. For example, you can upgrade to Version 5.3 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

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 5.3 rolling upgrade, all systems in the VAXcluster environment must be running at least Version 5.2 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 installation and operations guide for your VAX computer. If you have a VAXcluster environment, refer to the *VMS VAXcluster Manual*.

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

4.1 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

4.2 Cautions and Restrictions

You must be aware of the following cautions and restrictions before you begin the upgrade:

- 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.
- If you elect not to install VMS DECwindows software, the upgrade procedure removes existing VMS DECwindows files from the system disk. You can install VMS DECwindows software during the initial installation, or later using DECW$TAILOR.
- Do not attempt to use VMSINSTAL with OPTIONS G to transfer VMS and VMS DECwindows save sets to another media type. If you upgrade using a kit made in this manner, you will produce an unusable system.
- The VMS save sets must be in the [000000] directory on the system disk. The save sets are in this directory on all Digital-supplied distribution kits. If you have to move the VMS save sets, they must be in the [000000] directory, or the upgrade will fail.
- If you have a shared system dump file in SYS$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 *VMS Version 5.2 Release Notes*.
- You must not move the system disk or the VMS distribution kit from one drive to another during the upgrade.
- 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.
- 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.
- 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 5.3 of the VMS operating system is running, check the *VMS Version 5.3 Release Notes* for layered product restrictions. If problems persist, contact your Digital representative.

4.3 What You Need for the Upgrade

You need the following items to perform the upgrade to Version 5.3 of the VMS operating system:

- VMS Version 5.3 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 blank RX01 floppy diskette.

To prepare for an upgrade, follow the instructions in Chapter 5.

5 Before You Do an Upgrade

Before you do an upgrade, you need to perform several important tasks to prepare the system. The following list summarizes these tasks, each of which is described in this chapter:

1. Understanding VMS Version 5.3—You need to read the *VMS Version 5.3 New Features Manual* and *VMS Version 5.3 Release Notes* before you upgrade the system.
2. Saving AUTOGEN FEEDBACK parameters.
3. Backing up the system—For your protection, you must make a backup copy of the system disk and, in some cases, the console media.
4. Preparing the system disk—You need to move any files that might affect the upgrade to a user directory and make sure that you have enough space on the system disk for the upgrade. You also need to check and set certain SYSGEN parameters.
5. Preparing the system—You must stop all queues and, if applicable, shut down the network. You also need to set the system up for automatic restart. For some VAX computers this means modifying boot command procedures.

NOTE: If you are using volume-shadowed system disks, you must perform additional tasks before doing an upgrade. For more information, refer to the *VAX Volume Shadowing Manual*.

During the upgrade procedure, you are asked to boot the system from [SYSF]. Some VAX computers require a special command procedure to boot from [SYSF]. If you do not know how to boot from [SYSF], read Appendix A in this guide.

5.1 Understanding Version 5.3 of the VMS Operating System

✓ Before you begin the upgrade, read *VMS Version 5.3 New Features Manual* and *VMS Version 5.3 Release Notes*. These manuals describe the changes made to the VMS operating system in Version 5.3.

5.2 AUTOGEN FEEDBACK Parameters

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.

5.3 Backup Procedures

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 still will have a working system disk. To make a backup copy of the system disk, follow the instructions in the VMS installation and operations guide 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 installation and operations guide for your VAX computer.

5.4 Preparing the System Disk

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

1 Log into the SYSTEM account.

2 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 installation and operations guide for your VAX computer.

3 If applicable, remove the system disk and replace it with the backup copy.

4 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 installation and operations guide for your VAX computer.

5 Log into the SYSTEM account.

6 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.

7 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 5.2 Release Notes*.

8 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:

```
%VERIFY-I_OPENQUOTA, error opening QUOTA.SYS
```

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

Dec recommends Running this 2 to 3 times until no errors are received.

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

9 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 size (PAGEFILE.SYS) is greater than 4600 blocks, press RETURN. 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.

10 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
```

The number of free blocks should be greater than 31,000.

11 During the upgrade procedure you will shut down and reboot the system a number of times. To avoid configuring all the devices on the system during the reboot, set the SYSGEN parameter STARTUP_P1 to MIN. Enter the following commands, pressing RETURN after each one:

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

12 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.

13 To prepare the system for the upgrade, go to Section 5.5.

5.5 Preparing the System

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

1 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 2:

```
%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
```

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

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

3 If your computer uses console media, you can avoid rebooting the system manually during the upgrade, by setting 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 installation and operations guide 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 installation and operations guide for your VAX computer.

4 To activate any SYSGEN parameters that you changed (STARTUP_P1, for example), you must shut down and reboot the system at this point. 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 RETURN.

5 After the system reboots, log into the SYSTEM account.

Run SYSGEN to reconfigure the devices on the system and run STARTUP CONFIGURE. Enter the following commands:

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

6 To prevent users from logging into the system during the upgrade, enter the following command:

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

7 Perform this step only if you are upgrading one of the following VAX computers (otherwise go to Section 5.6):

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

Before beginning the upgrade procedure, you must modify the boot command procedures, DEFBOO and *ddd*GEN. 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 installation and operations guide 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 BOOT command, as follows:

```
BOOT dddn/R5=F0000000
```

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

```
DEPOSIT R5 F0000001
```

Go to Section 5.6.

8 Perform this step only if you are upgrading a VAX 8200, 8250, 8300, or 8350 computer (otherwise go to Section 5.6). Before beginning the upgrade, you must create a boot command procedure that allows you to boot from the [SYSF] directory.

Appendix A describes the procedures for setting the system up to boot from the [SYSF] directory.

Go to Section 5.6.

5.6 Summary

All of the following conditions must be met for the upgrade procedure to continue. Make sure:

- SYSUAF.DAT is in SYS$SYSTEM
- PAGEFILE.SYS is in SYS$SPECIFIC:[SYSEXE]
- The system disk has 31,000 free blocks
- Volume shadowing is disabled on the system disk

- STARTUP_P1 is set to MIN
- DECnet–VAX is shut down
- The queue manager is not running
- The page file is at least 4600 blocks
- The system is set to automatically restart

To upgrade a standalone system, go to Chapter 7.

To upgrade a VAXcluster environment, go to Chapter 6.

6 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 6.1.
- 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 6.2.

6.1 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 into 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 installation and operations guide 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 RETURN 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 7.

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 3 and 4 to upgrade each system disk in the cluster. When you finish upgrading each system disk, go to step 6.

6 Reboot each computer in the VAXcluster environment.

7 Then complete the post-upgrade procedure, described in Chapter 8.

6.2 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 5.3 rolling upgrade, the other nodes in the cluster must be running at least Version 5.2 of the VMS operating system.**

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 6.1.

6.2.1 Performing a Rolling Upgrade

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 RETURN 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 into the SYSTEM account on the VAX computer you are shutting down.

b. Enter the following command and press RETURN:

```
$ @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 7.

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 distibution media.

6 Reboot each VAX computer that boots from the system disk you have just upgraded.

7 Repeat steps 2 through 6 for each system disk you are upgrading.

8 Then complete the post-upgrade procedure, described in Chapter 8.

7 Standalone Upgrade

This chapter explains the upgrade procedure for a single VMS system. Before you begin the upgrade procedure, you must read the overview of the upgrade procedure described in Chapter 4 and complete the entire pre-upgrade procedure described in Chapter 5.

CAUTION: You cannot install or upgrade to Version 5.3 of the VMS operating system on the following VAX computers and system disks:

- **MicroVAX I, VAXstation I, and VAX 11/725**
- **RK07, RL02, RC25, 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 installation and operations guide for your VAX computer.

7.1 Beginning the Upgrade Procedure

This section describes how to get your disk and tape drives ready and 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 media in the source-drive. If your VMS distribution kit contains several pieces of media, put the first volume in the drive.
2. To start VMSINSTAL, use the following procedure:
   a. Log in to the SYSTEM account.
   b. Enter the following commands and press RETURN after each one:

```
$ SET DEFAULT SYS$UPDATE
$ @VMSINSTAL
```

   VMSINSTAL displays messages similar to the following:

```
        VAX/VMS Software Product Installation Procedure V5.2

It is 19-APR-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 installation and operations guide 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 7.1.

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 RETURN:

```
* 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 7.1.

5 When VMSINSTAL displays the following message, type VMS053 and press RETURN. Make sure that you use the number zero when you type VMS053. For example:

```
* Products: VMS053
```

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

```
* Enter the 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
```

After several minutes, the procedure displays messages similar to the following:

```
%MOUNT-I-MOUNTED, VMS053 mounted on _DJA2:
The following products will be processed:
  VMS V5.3
```

After several more minutes, the procedure displays messages similar to the following:

```
        Beginning installation of VMS 5.3 at 15:05
%VMSINSTAL-I-RESTORE, Restoring product saveset A...
                        VAX/VMS V5.3 Upgrade Procedure
                    Model: VAX-8800
              System device: RA82 - KROK$DUA1:
                Free blocks: 752667
```

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 (for YES) and press RETURN.

To interrupt the upgrade, do the following:

a. Type N (for NO) and press RETURN.

b. Press CTRL/Z.

9 The procedure displays the current date and time. If the date and time are correct, type Y (for YES). If they are incorrect, type N (for NO). 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-APR-1990 15:10
```

10 Continue with Phase 1 of the upgrade.

7.2 Upgrade Phase 1

This section describes Phase 1 of the upgrade. During Phase 1 you provide passwords for system accounts and select VMS and VMS DECwindows options. During Phase 1 the procedure restores the VMS Version 5.3 files in the VMS *required* save set. At the end of Phase 1, 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-APR-1990 15:11:17.05
```

1 To ensure system security, the upgrade procedure requires you to change the passwords for the SYSTEM, SYSTEST, and FIELD accounts before continuing. Passwords must be at least eight 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 the password for account FIELD 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-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 (for NO) and press RETURN. When the procedure asks if you want to enter a new password, type Y (for YES) 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
```

2 The procedure does the following:

- Turns off quotas on the system disk
- Stops OPCOM
- Stops the error formatter (ERRFMT)

3 If your VAX computer does not have console media, go to step 5. If your VAX computer has console media, the procedure might ask you to insert the console media in the drive. Use the backup copy of the console media. Make sure the console media is write-enabled. Insert the console media in the drive and do not remove it from the drive until the upgrade has been completed. The procedure displays a message similar to the following:

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

If you are booting directly from a local system disk, type N (for NO) and press RETURN. If you are using a CI750, type Y (for YES) and press RETURN.

If you typed Y (for YES), the procedure displays a series of messages about booting from console media. Follow the instructions displayed on the screen.

4 The procedure displays the following message:

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

To interrupt the upgrade, type N (for NO) and press RETURN.

To continue the upgrade, type Y (for YES) and press RETURN. For some computers, the procedure temporarily modifies 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 if you have a VAX 8600 computer) file that is set up to boot from the current system disk and system root.

5 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.

6 The procedure displays a series of messages about VMS and VMS DECwindows files. To select the VMS and VMS DECwindows options you want to install, follow the instructions displayed on the screen. For example:

```
The full VMS kit will be installed on this system disk.
The following components (with approximate sizes) will be provided:

    VMS library                 - 27500 blocks
    VMS optional                -  9000 blocks

    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)
```

CAUTION: If you elect not to install VMS DECwindows software, the upgrade procedure removes existing VMS DECwindows files from the system disk. If you do not install VMS DECwindows at this time, you can install it after the upgrade using DECW$TAILOR. Please note that for TK50 tape cartridges, it is quicker to install VMS DECwindows software during the upgrade than to tailor VMS DECwindows on after the upgrade. For more information about using DECW$TAILOR to install VMS DECwindows software, see Appendix F.

If you do not want to install VMS DECwindows software, type N (for NO) and go to step 7. 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, do the following:

a. Type Y (for YES) 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 44500 blocks. If you
want to perform a partial installation, you can choose to install one
or more of the following DECwindows components:

-DECwindows device support      -   9500 blocks
-DECwindows applications        -  11000 blocks
-DECwindows programming support -  24000 blocks

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

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

To install selected portions of the VMS DECwindows kit, type N (for NO) 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 the applications files.
- Applications files—VMS DECwindows provides a variety of applications including the following: Bookreader, Calculator, Calendar, Cardfiler, Clock, DDIF Document Viewer, 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 (100DPI) 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.
- 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.

c. 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 DECwindows options that you want to install.

The VMS DECwindows files are listed in Appendix C. Use DECW$TAILOR to remove the VMS DECwindows files you do not need. For more information on DECW$TAILOR, see the VMS installation and operations guide for your VAX computer.

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

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

8 After several minutes, the upgrade procedure indicates that it will shut down to reboot the partially upgraded VMS Version 5.3 system.

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 ddcu
```

For more information on booting your computer from the SYSF directory, see Appendix A.

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

```
VAX/VMS Version BC53-xxx Major version id = 1 Minor version id = 0
```

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

```
Continuing with VAX/VMS V5.3 Upgrade Procedure.

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

The procedure also displays messages about licenses which you can ignore.

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 you restore the VMS files, the procedure displays the following message:

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

Remove the VMS distribution media from the drive.

3 If you are not installing VMS DECwindows software, go to Section 7.4.

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

```
Put the DECwindows distribution kit in the drive.

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

Note: If you have either an RL02 or RK07 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 (for YES) and press RETURN.

4 The procedure displays messages similar to the following:

```
%MOUNT-I-MOUNTED, DECW53 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

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

5 Remove the VMS DECwindows distribution media from the drive.

7.4 Upgrade Phase 3

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

```
Continuing with VAX/VMS V5.3, Upgrade Procedure

     Upgrade Phase 3 19-APR-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.

7.5 Upgrade Phase 4

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

```
Continuing with VAX/VMS V5.3 Upgrade Procedure.

     Upgrade Phase 4  19-APR-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. You must 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 5), 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 RETURN after each one:

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

   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
   ```

   For more information on booting the system, see the installation and operations guide for your VAX computer.

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

```
VAX/VMS Version BC53-xxx Major version id = 1 Minor version id = 0
```

7.6 Upgrade Phase 5

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

```
Continuing with VAX/VMS V5.3 Upgrade Procedure.

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

During Phase 5, the upgrade procedure does the following:

- Displays messages about licenses which you can ignore.
- 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 5.3 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 8 explains these tasks.

7.7 Upgrade Phase 6

At the beginning of Phase 6, the procedure displays a message similar to the following:

```
Continuing with VAX/VMS V5.3 Upgrade Procedure.

     Upgrade Phase 6       19-APR-1990 18:09:38.14
```

During Phase 6, the upgrade procedure configures all devices on the system and runs AUTOGEN:

1 AUTOGEN runs and displays the following series of messages:

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

An attempt may be made to re-size 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 the procedure displays a message similar to the following, see the *Guide to Maintaining a VMS System* for information on setting the sizes of the PAGE, SWAP, and DUMP files.

```
******************
%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.
******************
```

2 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
    .
    .
    .
```

3 After AUTOGEN is complete, 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 installation and operations guide for you VAX computer.

4 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.

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

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

VAX/VMS V5.3

You have SUCCESSFULLY installed the VMS operating system.

The system is now executing the STARTUP procedure. Please
await the completion of STARTUP before logging into the
system (approximately three minutes).

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

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-APR-1990 13:00:00.00
 SYSTEM       job terminated at 19-APR-1990 13:00:00.00

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.

6 Press RETURN. The system asks for the user name and password. Log in to the SYSTEM account so that you can perform certain post-procedures procedures.

```
    Welcome to VAX/VMS V5.3
USERNAME: SYSTEM
PASSWORD:
    .
    .
    .
    Welcome to VAX/VMS V5.3
```

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

Chapter 8 explains the post-upgrade tasks you must complete.

8 After You Do an Upgrade

After you have upgraded 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, whether they are optional or required, and the order in which you perform them. The following list summarizes the tasks described in this chapter:

1 Replacing SYSUAF.DAT

2 Examining your command procedures

3 Changing MODPARAMS.DAT

4 Running DECnet–VAX

5 Updating your console media

6 Decompressing the system libraries

7 Testing the system with UETP

8 Removing unwanted files and checking the sizes of system files

9 Customizing and starting VMS DECwindows software

10 Building standalone BACKUP

11 Backing up the system disk

12 Running AUTOGEN

8.1 Replacing SYSUAF.DAT

If you permanently store SYSUAF.DAT on the system disk, skip this section and go to Section 8.2.

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.

8.2 Examining Your Command Procedures

The upgrade procedure places the latest versions of the following command procedures on your system disk:

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

Examine the old versions of these files before you purge them. Your old versions may have site-specific changes you want to save. For example, as part of your site-specific startup procedure you might install a page file on a disk other than the system disk. Edit the new versions of the files as necessary.

8.3 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.

8.4 Running DECnet–VAX

If you are not running DECnet–VAX, skip this section and go to Section 8.5.

If you are running DECnet–VAX, follow the instructions in Section 8.4.1 and Section 8.4.2.

8.4.1 Running NETCONFIG_UPDATE.COM

NETCONFIG_UPDATE.COM is a new command procedure 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 run this procedure whenever you want to change default access on your system.

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:

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 V5.3 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*.

8.4.2 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
            Userman SYSTEM will be used on nonlocal nodes
SYSMAN> DO @SYS$COMMON:[SYSMGR]UPDATE_CLUSTER_MEMBERS.COM
SYSMAN> EXIT
```

8.5 Updating Your Console Media

If your computer does not use console media, skip this section and go to Section 8.6.

If your computer uses console media, you must copy the Version 5.3 VMB.EXE onto your system's console media. Section 8.5.1 describes the procedure to use if you have a VAX 8530, 8550, 8700, 8800, 8810, or 8820-N.

Section 8.5.2 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

8.5.1 VAX 8530, 8550, 8700, 8800, 8810, 8820-N

If you have a VAX 8530, 8550, 8700, 8800, 8810, or 8820-N, 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 RETURN:

```
$ 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: V53CONSOLE
```

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
```

8.5.2 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
- Copies the new VMB 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 75 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      64   19-Apr-1990

Total of 1 file, 64 blocks.  Free space 23 blocks, largest 16.
```

If the size of the current VMB, plus the free space, does not equal at least 75 blocks, delete unneeded files on the console until you have 75 blocks available. In the above example, 87 blocks are available, so files do not have to be deleted.

3 Enter the following command and press RETURN:

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

4 If you are updating a VAX 8600 or 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.

8.6 Decompressing the System Libraries

Decompressing the system libraries gives the system faster access to the libraries. The decompressed libraries require approximately 8000 additional blocks of disk space. To find out how much disk space you have, enter the following command and press RETURN:

```
$ 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
```

8.7 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 9.

8.8 Increasing Free Space on the System Disk

Read this section if you want to remove some of the VMS operating system and DECwindows files that you do not need from the system disk and if you want to change the size of some system files.

8.8.1 Using VMSTAILOR

To remove unwanted VMS files from the system disk, use VMSTAILOR. For example, if you are not running DECnet–VAX, you do not need the network support files. To remove unwanted files with the VMSTAILOR program, do the following:

1 Log in to the SYSTEM account.

2 Enter the following command, and press RETURN:

```
$ RUN SYS$UPDATE:VMSTAILOR
```

VMSTAILOR displays the following message:

```
.
.
.
Do you want to tailor files "ON" or "OFF"?
```

3 Type OFF to remove unwanted files.

The VMSTAILOR program lists each group 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 group. A file group is made up of many small subgroups. You can eliminate an entire group of files, or you can eliminate one or more of its subgroups. For example:

```
CLASS - Network support
        Size of entire class (with subclasses): 1478
        Size of common files required for any subclass: 1233
        Do you wish to select the entire class (default = NO)?
```

4 Decide which file groups or subgroups you do not need to support your system. The VMSTAILOR program displays step-by-step instructions.

5 VMSTAILOR displays the names of the files it deletes. After it finishes, AUTOGEN runs automatically to make the adjustments that are necessary after system files are deleted. AUTOGEN also reboots the system.

Note: You can use VMSTAILOR at any time to delete or add groups of VMS files to the system disk. 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 5.3 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 5.3 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:VMSUn053.TXT.

8.8.2 Using DECW$TAILOR

To remove unwanted DECwindows files from the system disk, use DECW$TAILOR. For example, if you are not planning on utilizing the DECwindows programming support, you can remove those files. To remove DECwindows files from the system disk, do the following:

1 Log in to the SYSTEM account.

2 Enter the following command, and press RETURN:

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

DECW$TAILOR displays the following message:

```
.
.
.
Do you want to tailor files "ON" or "OFF"?
```

3 Type OFF to remove unwanted files.

The DECW$TAILOR program lists each group of files and its size in blocks. Files are grouped according to their function. For example, all the files required for VMS DECwindows user support are in one group. A file group is made up of many small subgroups. You can eliminate an entire group of files, or you can eliminate one or more of its subgroups. For example:

```
CLASS - DECwindows user support files
        Size of entire class (with subclasses): 8456
        No subclasses in this tailor class.
        Do you wish to select the entire class (default = NO)?
```

4 Decide which file groups or subgroups you do not need to support your system. The DECW$TAILOR program displays step-by-step instructions.

When tailoring VMS DECwindows files, consider the following:

- Applications files—VMS DECwindows provides a variety of applications; including the following: Bookreader, Calculator, Calendar, Cardfiler, Clock, DDIF Document Viewer, 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. You may also want these files on a nonworkstation VAX computer, because you can run VMS DECwindows applications on a nonworkstation VAX computer and display them on a workstation.
- 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.

Note: If you tailor device support files ON or OFF, the system reboots.

- 100 dots-per-inch (100DPI) 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.
- 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.

5 DECW$TAILOR displays the names of the files it deletes. If you deleted device support files, AUTOGEN runs automatically and reboots the system.

8.8.3 Checking the Size of System Files

You also can change the sizes of some files to increase free disk space. The size of the following files may have been changed to fit the system. Check these files to be sure that the sizes are appropriate.

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

To modify the size of these files, log into the SYSTEM account and enter the following command and press RETURN:

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

You might want to purge the following VMS files:

```
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:LOGIN.COM
SYS$MANAGER:SYSTARTUP_V5.COM
SYS$MANAGER:TFF$STARTUP.COM
SYS$MANAGER:WELCOME.TXT
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
```

8.9 Starting and Customizing VMS DECwindows Software

If you have installed VMS DECwindows software, there are several steps that you must take to start and customize the VMS DECwindows environment. Chapter 10 describes these steps. After you finish starting and customizing VMS DECwindows software, return to this chapter.

8.10 Backing Up the System Disk

Protect your work by making a backup copy of the upgraded system disk. Digital recommends that you also perform the following operations:

- Make a standalone BACKUP kit
- Back up the console media (if applicable)

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

8.11 Running AUTOGEN

Run AUTOGEN FEEDBACK and reboot the system after 24 hours of operation, and 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$FEEDBACK.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.*

9 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

9.1 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 9.2.

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 9.3).

If you encounter any problems in preparing to run UETP, read Section 9.3 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-APR-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 9.5.

If you want to run UETP without using the default responses, refer to Section 9.4 through Section 9.4.4, which explain the options.

9.2 Logging In

This section contains detailed instructions on running UETP. For a brief summary of these steps, see Section 9.1.

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.

9.2.1 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 9.3.2 discusses setting up scratch disks for testing.

9.3 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.

9.3.1 Setting Up 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.

9.3.2 Setting Up 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.

9.3.3 Setting Up Magnetic Tape Drives

Set up magnetic tape drives that you want to test by doing the following:

1 Place a scratch volume 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*.

9.3.4 Setting Up 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*.

9.3.5 Setting Up 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.

9.3.6 Preparing Ethernet Adapters for UETP Testing

Make sure that no other processes are sharing the device when you run UETP.

Note: UETP will not test your Ethernet adapter if DECnet–VAX or another application has the device allocated.

9.3.7 Preparing the DR11–W for UETP Testing

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.

9.3.8 Preparing the DRV11–WA for UETP Testing

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.

9.3.9 Preparing the DR750 or the DR780 for UETP Testing

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.

9.3.10 Preparing the MA780 for UETP Testing

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.

9.3.11 Preparing a Second LPA11–K for UETP Testing

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.

9.3.12 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.

9.3.13 Preparing for 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.

9.3.14 Preparing 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 to remove some files from the system disk before you run UETP. For instructions on using VMSTAILOR, see Chapter 3.

9.3.15 Preparing 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 chooses the node from which the DECnet–VAX test will be run. To choose the node to run the DECnet–VAX test on, 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 2.121
```

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
```

9.4 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 9.4.2 through Section 9.4.4. If you want to run all the test phases, skip the next section.

9.4.1 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]?
```

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 the next section.

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 9.4.2 and Section 9.4.4 discuss these questions. After you answer both questions, the phase you have selected runs to completion.

9.4.2 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 9.4.4). Make certain that all line printers and hardcopy terminals have enough paper, because each run requires two pages.

9.4.3 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 9.5.2.

9.4.4 Long and Short Report Format

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 9.5 for troubleshooting information.

9.4.5 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 9.4.2.)

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 9.6.

9.4.5.1 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.

9.4.5.2 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.

9.5 Troubleshooting

This section explains the role of UETP in interpreting operational errors in a VMS operating system. Section 9.5.4 discusses common errors that can appear in a UETP run and describes how to correct them.

9.5.1 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.

9.5.2 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 Volume* 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 V5.3

%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-APR-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.

9.5.2.1 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-APR-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
```

9.5.3 The 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.

9.5.4 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 Insufficient disk space

4 Incorrect VAXcluster setup

5 Problems during the load test

6 DECnet–VAX error

7 Errors logged but not displayed

8 No PCB or swap slots

9 Hangs

10 Bugchecks and machine checks

The following sections describe these errors and offer the best course of action for dealing with each one.

9.5.4.1 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 current account using the DCL commands SHOW PROCESS/PRIVILEGE and SHOW PROCESS/QUOTA as follows:

```
$ SHOW PROCESS/PRIVILEGES

19-APR-1990 18:06:02.89   OPA0:   User : SYSTEST

Process privileges :

    CMKRNL      may change mode to kernel
    CMEXEC      may change mode to exec
    SYSNAM      may insert in system logical name table
    GRPNAM      may insert in group logical name table
    DETACH      may create detached processes
    DIAGNOSE    may diagnose devices
    LOG_IO      may do logical I/O
    GROUP       may affect other processes in same group
    PRMCEB      may create permanent common event clusters
    PRMMBX      may create permanent mailbox
    SETPRV      may set any privilege bit
    TMPMBX      may create temporary mailbox
    NETMBX      may create network device
    VOLPRO      may override volume protection
    PHY_IO      may do physical I/O
    SYSPRV      may access objects via system protection

$ SHOW PROCESS/QUOTAS

19-APR-1990 18:06:03.36          OPA0:        User: SYSTEST

Process Quotas:

    Account name: SYSTEST
    CPU limit:                      Infinite    Direct I/O limit:     55
    Buffered I/O byte count quota:    32768     Buffered I/O limit:   18
```

```
Timer queue entry quota:          20    Open file quota:     100
Paging file quota:             19543    Subprocess quota:      8
Default page fault cluster:       64    AST limit:            98
Enqueue quota:                   300    Shared file limit:     0
Max detached processes:            0    Max active jobs:       0
```

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-APR-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.

9.5.4.2 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-APR-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-APR-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-APR-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 9.3.

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. Because either DECnet–VAX or the LAT terminal server might also want to use the Ethernet adapter (a shareable device), UETP requires that you shut down DECnet–VAX and the LAT terminal server before you run the device test phase.

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 9.4.1). 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.

9.5.4.3 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-APR-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-APR-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 3 for more information.

See Section 9.2.1 and Section 9.3.2 for a further discussion of disk space.

9.5.4.4 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 9.3.13 and Section 9.5.2.1 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 9.3.13 for information on preparing for VAXcluster testing.

9.5.4.5 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 9.6.3).

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-APR-1990 15:45:08.97
%UETP-I-BEGIN, UETLOAD02_0000 beginning at 19-APR-1990 15:45:09.42
%UETP-I-BEGIN, UETLOAD03_0001 beginning at 19-APR-1990 15:45:09.63
%UETP-I-BEGIN, UETLOAD04_0002 beginning at 19-APR-1990 15:45:10.76
%UETP-I-BEGIN, UETLOAD05_0003 beginning at 19-APR-1990 15:45:11.28
%UETP-I-BEGIN, UETLOAD06_0004 beginning at 19-APR-1990 15:45:12.56
%UETP-I-BEGIN, UETLOAD07_0005 beginning at 19-APR-1990 15:45:13.81
%UETP-I-BEGIN, UETLOAD08_0006 beginning at 19-APR-1990 15:45:14.94
%UETP-I-BEGIN, UETLOAD09_0007 beginning at 19-APR-1990 15:45:16.99
%UETP-I-BEGIN, UETLOAD10_0008 beginning at 19-APR-1990 15:45:19.32
%UETP-I-BEGIN, UETLOAD11_0009 beginning at 19-APR-1990 15:45:19.94
%UETP-I-BEGIN, UETLOAD02_0010 beginning at 19-APR-1990 15:45:20.20
%UETP-I-BEGIN, UETLOAD03_0011 beginning at 19-APR-1990 15:45:21.94
%UETP-I-BEGIN, UETLOAD04_0012 beginning at 19-APR-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.

9.5.4.6 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 authorization key (see the *VMS 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 9.4.1) to determine whether the error can be re-created.
- Refer to the *VMS System Messages and Recovery Procedures Reference Volume*.

9.5.4.7 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.

9.5.4.8 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-APR-1990 07:47:16.50
%UETP-I-BEGIN, UETLOAD02_0000 beginning at  19-APR-1990 07:47:16.76
%UETP-I-BEGIN, UETLOAD03_0001 beginning at  19-APR-1990 07:47:16.92
%UETP-I-BEGIN, UETLOAD04_0002 beginning at  19-APR-1990 07:47:17.13
%UETP-I-BEGIN, UETLOAD05_0003 beginning at  19-APR-1990 07:47:17.35
%UETP-I-BEGIN, UETLOAD06_0004 beginning at  19-APR-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-APR-1990 07:47:54.10
-UETP-W-TEXT, Aborted via a user CTRL/C.
 ************************************************
 *                                              *
    END OF UETP PASS 1 AT  19-APR-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.

9.5.4.9 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—Reboot the system with a larger value for NPAGEVIR.
- 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.

9.5.4.10 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.

9.6 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 9.4 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.

9.6.1 Initialization Phase

The following occurs during the initialization phase:

- The image UETINIT00.EXE prompts you for information (see Section 9.4). 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.

9.6.2 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 9.4.1.

9.6.2.1 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.

9.6.2.2 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 9–1 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
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

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-APR-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).

Table 9–1 lists the device test images and the devices to be tested.

Table 9–1 The 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
UETUNAS00.EXE	Ethernet Adapters

9.6.3 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 9.4.3) 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 9.4.1).

9.6.4 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 9.4.1.

9.6.4.1 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 test 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.

9.6.4.2 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-APR-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-APR-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

9.6.5 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 9.3.13.
- 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 9.5.4.1.
- 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.

10 Starting and Customizing VMS DECwindows Software

This chapter describes the tasks that you that you must perform to start and customize VMS DECwindows software. If you have not installed VMS DECwindows software on your system, skip this chapter. Depending on your configuration, VMS DECwindows startup and customization tasks include the following:

- Starting the VMS DECwindows software as described in Section 10.1.
- Customizing the server startup as described in Section 10.2.
- Using VMS/ULTRIX Connection software as a DECwindows transport.
- Using customer-written transports with DECwindows.
- Defining print formats as described in Section 10.6.
- Compiling the optional Ada software as described in Section 10.7.

10.1 Starting the VMS DECwindows Software

If you installed VMS Version 5.3 on your system, go to Section 10.1.1.

If you upgraded your system to VMS Version 5.3, go to Section 10.1.2.

10.1.1 After an Installation

After you register the VMS license and reboot the system, DECwindows starts automatically (if you installed it). However, if you intend to use DECnet–VAX you need to edit SYS$SYSROOT:[SYSMGR]SYSTARTUP_V5.COM and do the following:

1 Choose one of the following commands to start the network by removing the appropriate comment delimiter (!):

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

2 Add a comment delimiter (!) to the following command:

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

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.

10.1.2 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 this 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, including those that are not using VMS DECwindows.

10.1.3 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 10–1 lists the logical names contained in this table.

Table 10–1 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.
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.

10.2 Customizing the Server Startup

If you are not planning on starting the VMS DECwindows server, go to Section 10.6.

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 10–2. 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 10–2 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
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

(continued on next page)

Table 10–2 (Cont.) DECwindows Keymap

Dialect	Model	DECwindows Keymap Name
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 doesn't 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 10–3 lists the logical names.

Note: These logical names may change in a future release.

Table 10–3 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.
DECW$SERVER_SCREENS	Device name used by the server. The first two characters indicate the shareable image to be loaded for device support (for example, GCA0:).
DECW$SERVER_TRANSPORTS	A list of transport types that the DECwindows server will load when it starts up.
DECW$XSIZE_IN_PIXELS	Specifies the size of the workstation screen.
DECW$YSIZE_IN_PIXELS	Specifies the size of the workstation screen.

10.3 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.*

10.4 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 Digital supplied transports)
DECW_TRANSPORT_name (for customer-written transports)

Install the image as a protected image in the SYS$SHARE directory (as defined by an executive mode logical name). Digital recommends that you use the install the image by adding the following line to SYS$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 10.2.

10.5 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 10.2 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 (QUOTE>(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
```

10.6 Defining Print Formats

If you do not want to define print formats on your system, go to Section 10.7. 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 10–4 lists the logical names and the print formats associated with each.

Table 10–4 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 be overridden by users.

If you have any of the logical names in Table 10–4 defined in your environment, 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"
```

10.7 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 only to do this 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.

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 installation and operations guide 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 RETURN:

```
>>> B/F0000000 device-name
```

Substitute the device name of the system disk for *device-name*.

A.2 VAX 8530, 8550, 8700, 8810-N, 8820, 8830, and 8840

To boot from [SYSF], use the following procedure:

1 Press CTRL/P. On the VAX 8530, enter the following command at the console-mode prompt (>>>) and press RETURN:

```
>>> HALT
```

On the VAX 8820, enter the following command at the PS-OS-0> prompt and press RETURN:

```
PS-OS-0> HALT/CPU=ALL
```

2 On the VAX 8530, 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, 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 RETURN 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 RETURN:

```
$ 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 RETURN:

```
$ 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 RETURN:

```
>>> 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 RETURN 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 RETURN:

```
$ 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 RETURN:

```
$ 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 RETURN:

```
>>> 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 RETURN:

   ```
   >>> BOOT/R5:F0000000/XMI:3/BI:2/NODE:0E02 DU1
   ```

A.6 VAXstation 8000

To boot from [SYSF], use the following procedure:

1. Press CTRL/P to obtain the console prompt.
2. Enter the boot command in the following format:

   ```
   >>> B/R5:F0000000 [ddnu]
   ```

   Substitute the boot name of the system disk for *ddnu*. If you do not specify the boot name of the system disk, the VAXstation 8000 boots from the disk designated in EEPROM.

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.

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	PROG
	ALLOBJS.H	Demo source	PROG
	BITMAP.C	Demo source	PROG
	BITMAP.EXE	Demo source	PROG
	CLOCK.DDIF	Paint example	ALL
	DECBURGER.ADA	UIL DRM Demo source	PROG
	DECBURGER.C	UIL DRM Demo souce	PROG
	DECBURGER.COM	UIL DRM Demo file	PROG
	DECBURGER.EXE	UIL DRM Demo image	PROG
	DECBURGER.FOR	UIL DRM Demo file	PROG
	DECBURGER.UID	UIL DRM Demo file	PROG
	DECBURGER.UIL	UIL DRM Demo file	PROG
	DECBURGER_DEFS.FOR	UIL DRM Demo file	PROG
	DECW$TRANSPORT_ EXAMPLE.EXE	Demo image	PROG
	DIALOG.C	Demo source	PROG
	DWTVMSPUSHTEST.B32	XUI toolkit demo	PROG
	DWTVMSPUSHTEST.EXE	Demo image	PROG
	HELLOWORLD.ADA	XUI toolkit demo source	PROG
	HELLOWORLD.C	XUI toolkit demo	PROG

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	HELLOWORLD.EXE	XUI toolkit demo image	PROG
	HELLOWORLD.FOR	XUI toolkit demo	PROG
	HELLOWORLD.UID	XUI toolkit demo file	PROG
	HELLOWORLD.UIL	XUI toolkit demo file	PROG
	ICO.C	Demo source	PROG
	ICO.EXE	Demo image	PROG
	ICOPAS.PAS	Demo source	PROG
	OBJCUBE.H	Demo source	PROG
	OBJICO.H	Demo source	PROG
	OBJTETRA.H	Demo source	PROG
	PLAID.C	Demo source	PROG
	PLAID.EXE	Demo image	PROG
	POLYINFO.H	Demo source	PROG
	STRINGS.MAR	Low-level DECwindows utility routine	PROG
	XLIBINTRO.ADA	Demo source	PROG
	XLIBINTRO.C	Demo source	PROG
	XLIBINTRO.EXE	Demo image	PROG
	XLIBINTRO.FOR	Demo source	PROG
	XPORTEXAMPLEDEF.R32	Demo source	PROG
	XPORT_EXAMPLE.B32	Demo source	PROG
	XPORT_EXAMPLE_QUEUE.MAR	Demo source	PROG
	XPORT_EXAMPLE_XFER.MAR	Demo source	PROG
DECW$FONT:	See *VMS DECwindows Xlib Programming Volume*	DECwindows fonts	WS
DECW$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

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	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
	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

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	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
	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

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	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$MAIL_MAIN.UID	DECwindows VMSmail support file	ALL
	DECW$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

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	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
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.*[1]	DDIF converter language binding	PROG
	CDA$DTIF_TO_DDIF.EXE	CDA DTIF to DDIF converter image	ALL
	CDA$MSG.*[1]	DDIF message symbols	PROG
	CDA$WRITE_ANALYSIS.EXE	CDA image	ALL
	DDIF$DDIF_.ADA	DDIF Ada binding	PROG
	DDIF$DEF.*[1]	DDIF language binding	PROG
	DDIF$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

[1]Asterisk (*) indicates that BASIC, BLISS, FORTRAN, C, Macro, Pascal, and PLI languages are supported.

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	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$DWTDEF.*[1]	Widget include files	PROG
	DECW$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$DWTWIDGETDEF.*[1]	Widget include files	PROG
	DECW$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$SERVER_DIX.EXE	Server device independent support	WS
	DECW$SERVER_EXTENSION_ PEX.EXE	PHIGS X Server Extension	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

[1]Asterisk (*) indicates that BASIC, BLISS, FORTRAN, C, Macro, Pascal, and PLI languages are supported.

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	DECW$X_.ADA	Xlib Ada binding	PROG
	DECW$XLIBDEF.*[1]	Language Xlib binding	PROG
	DECW$XLIBMSG.*[1]	Language message symbols	PROG
	DECW$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$GB_UCODE.EXE	PHIGS image	WS
SYS$MANAGER:	DECW$CHECK_PARAMS.COM	Check AUTOGEN parameters	ALL
	DECW$DEVICE.COM	Device detection and configuration	ALL
	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

[1]Asterisk (*) indicates that BASIC, BLISS, FORTRAN, C, Macro, Pascal, and PLI languages are supported.

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	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
	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$MASTER.EXE	FileView image	ALL
SYS$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

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
	GCBDRIVER.EXE	VAXstation 2000 monochrome 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
	DECW$TAILOR_ON.TEMPLATE	Tailoring support file	ALL

(continued on next page)

Table C–1 (Cont.) DECwindows Software Components

Directory	File	Purpose	Component Class
VUE$LIBRARY:	VUE$*.COM	FileView verb command files	ALL
	VUE$SYSTEM_PROFILE.VUE$DAT	FileView default system profile	ALL

D Layered Products Caution

Because of the way the VMS Version 5.3 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 5.3 is running, check the *VMS Version 5.3 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 5.3 as of October 12, 1989. 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.

Table D–1 Layered Products Currently Available for VMS Version 5.3

Product Name	Most Recent Version Number
3270 Terminal Option Software	1.0
AAF01/VMS Subroutine Library	2.0
ADF01/VMS Subroutine Library	4.0
ALL-IN-1	2.3
ALL-IN-1 Starter	1.0
ALL-IN-1 System for Sales and Marketing	1.2
Baseview	1.2
C.A.S. Delivery System	1.6
CDA Converter Library for VMS	1.0
CMR21 Host Utility	1.1
Courseware Authoring System	1.6
Courseware Design System	1.6
DEC Capacity Planner for VMS	1.0
DEC GKS for VMS	4.0
DEC PHIGS for VMS	2.0
DECdecision	1.0
DECdx/VMS	1.2
DECforms	1.1
DECimage Application Services for VMS	2.0
DECimage Scan Software for VMS	2.0
DECintact	1.1

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.3

Product Name	Most Recent Version Number
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	5.3
DECpage	3.0
DECrouter 200	1.1
DECrouter 2000	1.1
DECserver 100 for VMS and MicroVMS	2.0
DECserver 200 for VMS and MicroVMS	3.0
DECserver 300 for VMS	1.0
DECserver 500 for VMS	2.0
DECtalk Mail Access	1.1
DECvoice Software	1.0
DECwindows DECnet/SNA 3270 Terminal Emulator for VMS	1.0
DECwrite for VMS	1.0
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	4.0
Internet Portal	1.0
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	1.1
LCG01 Software	1.5

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.3

Product Name	Most Recent Version Number
MicroVAX VSV21 Support Software	3.0
MicroVAX/DRQ3B Device Driver	1.2
MicroVAX MIRA Switch Control	2.1
MUXserver 100 Remote Terminal Server	1.0
MUXserver 300 Remote Terminal Server for VMS	1.0
NMCC/DECnet Monitor	2.1
NMCC/VAX ETHERnim	2.1
PBXserver	2.1
PCLAN/Server VMS	2.2
PDP–11 DATATRIEVE/VAX	3.3
PDP–11 FORTRAN–77/VAX to RSX	5.3
PDP–11 Symbolic Debugger/VAX to RSX	2.1
PLXY-11/VAX	1.4
ReGIS Software	1.1
Remote Bridge Management Software	1.2
Remote System Manager	2.2
Session Support Utility	1.1
Terminal Server Manager	1.3
ULTRIX Mail Connection	1.0
VAX 2780/3780 Protocol Emulator	1.7
VAX 3271 Protocol Emulator	2.4
VAX ACMS	3.1
VAX Ada	2.0
VAX ADE	2.4
VAX APL	3.1
VAX BASIC	3.3
VAX BCP	1.1
VAX BLISS-32 Implementation Language	4.6
VAX C	3.0
VAX CDD/Plus	4.1
VAX COBOL	4.2
VAX COBOL Generator	1.3
VAX Computer Integrated Telephony Applications Interface	1.1
VAX Computer Integrated Telephony PBX Server	1.1
VAX CoProcessor/RSX	1.0
VAX DAL	1.6

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.3

Product Name	Most Recent Version Number
VAX Data Distributor	2.0-1
VAX DATATRIEVE	4.2
VAX DBMS	4.1
VAX DEC/CMS	3.2
VAX DEC/MAP	2.1
VAX DEC/MMS	2.5
VAX DEC/Shell	2.1
VAX DEC/Test Manager	3.0
VAX DECalc	3.0
VAX DECalc-Plus	3.0
VAX DECalc/DECgraph Package	3.0
VAX DECgraph	1.5
VAX Decision Expert for VMS	1.0
VAX DECrad	3.2
VAX DECscan VMS and ELN Bitbus Drivers	2.1
VAX DECscan VMS Software Toolkit	2.0
VAX DECslide	1.3
VAX DECspell Verifier/Corrector	1.1
VAX DIBOL	4.0
VAX Distributed File Service	1.2
VAX Distributed Name Service	1.1
VAX Distributed Queuing Service	1.1
VAX DOCUMENT	1.2
VAX DSM	5.1
VAX DT07	3.0
VAX DY32	3.0
VAX EDCS	1.2
VAX Encryption	1.1
VAX FMS	2.4
VAX FORTRAN	5.3
VAX FTAM	1.1
VAX Grammar Checker	1.0
VAX Jukebox Control Software	1.0
VAX KCT32	2.0
VAX Key Distribution Center	1.1
VAX KMS11-BD/BE HDLC/BSC Framing Software	2.0

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.3

Product Name	Most Recent Version Number
VAX KMS11-BD/BE X.25 Link Level Software	2.0
VAX Language Sensitive Editor	2.3
VAX LIMS/SM	1.4
VAX LISP for VMS Systems	3.0
VAX LN03 Image Support Software	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	2.0
VAX OPS5	3.0
VAX OSI Application Kernel	1.1
VAX Packetnet System Interface	4.2
VAX Packetnet System Interface Access	4.2
VAX Pascal	3.9
VAX PBX/Facilities Management	3.0
VAX Performance Advisor	2.0
VAX Performance and Coverage Analyzer	2.1
VAX PL/I	3.3
VAX PrintServer Client Software	3.0
VAX PrintServer Supporting Host Software	3.1
VAX Public Access Communications	1.2
VAX Rally	2.1
VAX Rdb/ELN	2.2
VAX Rdb/VMS	3.0
VAX ReGIS to Sixels Converter	1.0
VAX Remote Environmental Monitoring Software	1.1
VAX RMS Journaling	5.3
VAX RPG II	2.1
VAX Scan	1.1
VAX ScriptPrinter Software	2.1
VAX Software Project Manager	1.2
VAX Source Code Analyzer	1.3
VAX Software Performance Monitor	3.4
VAX SQL/Services	1.0

(continued on next page)

Table D–1 (Cont.) Layered Products Currently Available for VMS Version 5.3

Product Name	Most Recent Version Number
VAX Storage Library System	2.0
VAX TDMS	1.8
VAX TEAMDATA	1.3
VAX TU70/72 Device Driver	1.2
VAX Volume Shadowing	5.3
VAX VTX	4.0
VAX Wide Area Network Device Drivers	1.1
VAX XWAY	1.1
VAX–11 RSX	2.4
VAXcluster Console System	1.2
VAXcluster Software	5.3
VAXELN Ada	2.0
VAXELN Toolkit	4.0
VAXLAB Software Library	1.3
VAXlink for IMS and VSAM Client	1.0
VAXpc for VMS	1.0
VAXset	8.0
VIDA for DB2	1.0
VIDA with IDMS/R	2.2
VMS Workstation Software	4.2
VMS/SNA	2.0
VMS/ULTRIX Connection (tm)	1.0
WPS-PLUS/VMS	3.1
X25Portal 2000	1.0
X25Router 2000	1.0

E Release Notes for the VAX 8200

If you are installing the VMS operating system on a VAX 8200 from an HSC disk drive, read this section before you perform the installation.

E.1 How to Proceed

If you have edited the CIBOO.CMD file on the console RX50 to reflect your system configuration, you can use the BOOT58 program and a modified version of CIBOO.CMD (DEFBOO.CMD) to boot the VMS operating system automatically from an HSC disk drive. To install the VMS operating system, follow the instructions in Chapter 2.

If you have not edited the CIBOO.CMD file to reflect your system configuration, the BOOT58 program displays syntax error messages when you try to boot the VMS operating system automatically. You must deposit values at the *BOOT58>* prompt to boot the VMS operating system manually from an HSC disk drive. If you have not edited CIBOO.CMD, install the VMS operating system as described in Section E.2.

Once you have installed and booted the VMS operating system and are logged in to the SYSTEM account, you can edit CIBOO.CMD. Then you can use a modified version of CIBOO.CMD (DEFBOO.CMD) to perform subsequent boots.

E.2 Installing the VMS Operating System

If you have not edited CIBOO.CMD, follow this procedure to install the VMS operating system from an HSC drive.

Prepare for the installation as described in the *VMS Installation and Operations: VAX 8200, 8250, 8300, 8350*. Obtain the following information from either the network or VAXcluster manager:

- VAXBI node number
- HSC node number
- HSC controller number or numbers
- Unit number of the disk drive

Then boot standalone BACKUP as follows:

1. Identify the disk drive that contains the distribution disk and the HSC that controls it. You need to know the node number of the HSC and the unit number of the disk drive.
2. Make sure the console RX50 is in the console diskette drive, CSA1.
3. To start the BOOT58 program, enter the following command and press RETURN:

```
>>> B/R5:800 CSA1
```

4 At the *BOOT58>* prompt, deposit the following values. Note that all numeric entries are made using hexadecimal notation.

a. Deposit the value 20 for the type code for the boot device for the HSC on a CIBCI or CIBCA device into register 0 using the following command:

```
BOOT58> D/G 0 20
```

b. Deposit the VAXBI node number into register 1 using the following format:

```
BOOT58> D/G 1 node-number
```

For example, if the VAXBI is node 6, enter the following and press RETURN:

```
BOOT58> D/G 1 6
```

c. Deposit the HSC node number into register 2 using the following format:

```
BOOT58> D/G 2 node-number
```

For example, if the HSC is node 12 on a CIBCI device, enter the following command and press RETURN:

```
BOOT58> D/G 2  C
```

Note: If the drive holding the system disk is accessible to two HSCs, deposit both node numbers. Put the greater number in hexadecimal digits 3 and 2, and the lesser in digits 1 and 0. For example, suppose one HSC is numbered 18 (hexadecimal 12) and the other is numbered 10 (hexadecimal A), enter the following and press RETURN:

```
BOOT58> D/G 2 12OA
```

d. Deposit the unit number of the drive holding the system disk into register 3 using the following format:

```
BOOT58> D/G 3 unit-number
```

For example, if the drive holding the system disk is unit number 21, deposit hexadecimal 15 into register 3:

```
BOOT58> D/G 3 15
```

e. Deposit the number of the correct root directory from which to boot using the following format (where *r* is the directory root):

```
BOOT58> D/G 5 r0000000
```

By default, the VMS operating system is in SYS0. For example, if the correct root directory is SYS0, enter the following and press RETURN:

```
BOOT58> D/G 5 0
```

f. Enter the following commands and press RETURN after each one:

```
BOOT58> D/G 4 0

BOOT58> D/G E 200

BOOT58> LOAD VMB.EXE/START:200

BOOT58> START 200
```

5 The procedure may ask for the date and time. Enter the date and time using the 24-hour clock format and press RETURN. For example:

```
VAX/VMS Version V5.3 Major version id = 01 Minor version id = 00

PLEASE ENTER DATE AND TIME (DD-MMM-YYY HH:MM:) 19-APR-1990 13:00
```

6 The procedure displays a list of the local devices on your system and, if you have them, HSC- and MSCP-served devices. For example:

```
Available device DJA2    device type RA60
Available device DJA3    device type RA60
```

7 Check the list of devices. If the list is incomplete, make sure that all the drives are connected properly to the system. See your hardware manuals for details.

8 When standalone BACKUP finishes booting, it displays an identification message followed by the dollar sign prompt ($):

```
%BACKUP-I-IDENT, Stand-alone BACKUP V5.3; the date is 19-APR-1990 13:00:00.00)
$
```

Continue installing the VMS operating system as described in Chapter 2.

E.3 Editing CIBOO.CMD

Once you have installed and booted the VMS operating system and are logged in to the SYSTEM account, edit the CIBOO.CMD file using the following procedure:

Note: If you are editing CIBOO.CMD on a hardcopy terminal you must use line mode edit. You might want to use a video display terminal instead.

1 Make sure the console RX50 is in the console diskette drive, CSA1.

2 To connect the console drive to the system, enter the following commands and press RETURN after each one:

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> CONNECT CONSOLE
SYSGEN> EXIT
```

3 Use the Exchange Utility to copy CIBOO.CMD to the system disk. Enter the following command and press RETURN:

```
$ EXCHANGE COPY CSA1:CIBOO.CMD *.*
```

4 To edit CIBOO.CMD, enter the following command and press RETURN:

```
$ EDIT CIBOO.CMD
```

5 CIBOO.CMD contains the following text:

```
!CIBOO.CMD   :Boot command file to boot a VAX 8200/8300 from an HSC disk.
!
!
!Note "n", "p" (and "q"), "u", and "r" are single hexadecimal characters
!
D/G 0 20                ! CI Port Device Type Code
!D/G 1 n                ! n = CI adaptor's VAXBI node number
!D/G 2 p                ! Use the HSC controller at CI node p
!D/G 2 0p0q             ! Use either the HSC controller at CI nodes p and q
!D/G 3 u                ! u = Disk drive unit number
D/G 4 0                 ! Boot Block LBN (not used)
!D/G 5 r0000000         ! r = system root [SYSR...], Software boot flags
D/G E 200               ! Address of Working Memory+^X200
LOAD VMB.EXE/START:200  ! Load Primary Bootstrap
START 200               ! Start Primary Bootstrap
```

a. Delete the comment character (!) that appears before the D/G 1 command and replace *n* with the VAXBI node number of the CI adapter.

b. If your computer is only connected to one HSC controller, delete the comment character (!) that appears before the first D/G 2 command. Replace *p* with the HSC controller number in hexadecimal notation. Delete the second D/G 2 command.

   If the drive holding the system disk is connected to two HSC controllers, do the following:

   - Delete the comment character (!) that appears before the second D/G 2 command
   - Replace *p* with the controller number of the first HSC in hexadecimal notation
   - Replace *q* with the controller number of the second HSC in hexadecimal notation
   - Delete the first D/G 2 command

c. Delete the comment character (!) that appears before the D/G 3 command and replace *u* with the unit number of the HSC drive from which you will boot the VMS operating system. Use hexadecimal notation.

d. Delete the comment character (!) that appears before the D/G 5 command and replace *r* with the number of the correct root directory from which to boot. By default, the VMS operating system is in SYS0.

e. Exit from the editor.

6 Rename the file. Enter the following command and press RETURN:

```
$ RENAME CIBOO.CMD DEFBOO.CMD
```

7 Use the Exchange Utility to copy DEFBOO.CMD to the console RX50. Enter the following command and press RETURN:

```
$ EXCHANGE COPY DEFBOO.CMD CSA1:DEFBOO.CMD
```

8 When you are finished, enter the following command and press RETURN:

```
$ DISMOUNT CSA1
```

9 To secure the console from unauthorized access, you must enter the following command and press RETURN:

```
$ MOUNT/FOREIGN/SYSTEM/NOWRITE/NOASSIST CSA1
```

If you change the system configuration, use the previous procedure to modify CIBOO.CMD.

F Using DECW$TAILOR to Install DECwindows

The VMS upgrade and installation procedures give the opportunity to install VMS DECwindows software. If you decide you want install VMS DECwindows software *after* a VMS installation or upgrade, use DECW$TAILOR as follows:

1. Log in to the SYSTEM account.
2. Enter the following command, and press RETURN:

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

   DECW$TAILOR displays the following message:

   ```
      .
      .
      .
   Do you want to tailor files "ON" or "OFF"?
   ```

3. Type ON and press RETURN.
4. Follow the instructions given in the screen displays. The DECW$TAILOR program is intended to be self-explanatory.

When tailoring VMS DECwindows files, consider the following:

- Applications files—VMS DECwindows provides a variety of applications; including the following: Bookreader, Calculator, Calendar, Cardfiler, Clock, DDIF Document Viewer, 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. You may also want these files on a nonworkstation VAX computer, because you can run VMS DECwindows applications on a nonworkstation VAX computer and display them on a workstation.
- 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.

Note: If you tailor device support files ON or OFF, the system reboots.

- 100 dots-per-inch (100DPI) 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.

- 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 Massachusetts Institute of Technology.

Index

A

Ada
  compiling source files • 10–9
  programming interfaces • 10–9
AUTOGEN
  at end of phase upgrade • 7–12
  FEEDBACK data • 5–1
AUTOGEN.PAR
  creation of • 7–10

B

Boot command procedures
  See DEFBOO and dddGEN
Bugcheck message
  during UETP • 9–27

C

CIBOO.CMD • E–1
  editing • E–3
  example • E–4
Cluster upgrade
  See Concurrent upgrade
  See Rolling upgrade
Command procedures
  processed during upgrade • 8–2
Concurrent upgrade
  description • 6–1
  pre-upgrade procedure • 6–1 to 6–2
  procedure • 6–1 to 6–2
  summary • 6–1
Console media
  needed for an installation • 2–2
CTRLNAME logical name • 9–8, 9–29

D

dddGEN
  restoring original versions during upgrade • 7–10
DECnet–VAX
  configuration database • 8–2
  error message during UETP • 9–25
  preparing for UETP • 9–10
  registering the license • 3–1
  requirement for upgrade • 5–5
  security measures • 8–2
  UETP test of • 9–33
  UETP test phase • 9–32
DECW$TAILOR
  adding files to a system disk • 8–9
  removing files from a system disk • 3–4, 8–9
  using to install DECwindows • F–1
DECwindows
  See VMS DECwindows
DEFBOO
  restoring original versions during upgrade • 7–10
Device
  format for upgrade • 7–2
  HSC name format for upgrade • 7–2
Device test (in UETP)
  running individually • 9–29
Diagnostics
  relationship to UETP • 9–14
Digital Small Storage Interconnect (DSSI) disks
  device name change • 2–4
Directories
  user, processed during upgrade • 7–9
Disk
  See System disk, User disk
Distribution volume
  use during upgrade • 7–1

E

Error during UETP
  diagnosing • 9–14
  sources of • 9–15

Error during upgrade
  cluster hang during shutdown • 6–3
  during console update • 8–6
  while entering device code • 7–2
Error Log Utility
  relationship to UETP • 9–2, 9–14, 9–25
Ethernet
  defining a remote node for UETP • 9–16
  preparing for UETP • 9–6
EXPECTED_VOTES parameter • 8–2

H

Hang
  See System hang
HSC (Hierarchical Storage Controller)
  device format for an installation • 2–6
  device format for upgrade • 7–2
  displaying name • 2–3

I

Installation procedure
  definition • 1–1
  for VAX 8200 • E–1 to E–5
  stages of • 2–2
Interrupt
  upgrade before Phase 1 • 7–3

K

Keyboard layout • 10–3
  customizing • 10–3
  list of • 10–4
Keyboard model number • 10–3
Keymap name
  determining • 10–3

L

License registration
  for system integrated products • 3–1
  for VMS • 3–1
Line printer
  preparing for UETP • 9–2, 9–4, 9–6
  UETP output • 9–30
  UETP test image • 9–31
  UETP test of • 9–28
LOADS logical name • 9–32
Load test
  defining user load for UETP • 9–12
  description • 9–31
  error during UETP • 9–24
  running individually • 9–11
Local Area VAXcluster, upgrade requirement • 4–3
Log file generated by UETP
  See also UETP.LOG
  during the load test • 9–24
  NETSERVER.LOG • 9–23
  OLDUETP.LOG • 9–17
Logical names • 10–6
  for DECwindows • 10–2
Logical name used by UETP
  CTRLNAME • 9–29
  LOADS • 9–32
  SYS$INPUT • 9–29
  SYS$OUTPUT • 9–30

M

Magnetic tape
  preparing for UETP • 9–2, 9–4, 9–5
  test of • 9–28, 9–30
  UETP test image • 9–31
Master command procedure
  See UETP.COM
Mixed-interconnect configuration, upgrade
    requirement • 4–3
Mixed-version cluster
  upgrading to • 6–2 to 6–3
MODE • 9–36
MODE logical name • 9–15
MODPARAMS.DAT
  modifying for cluster • 8–2
  modifying for single system • 8–2
Monochrome workstation • 10–3

N

NETCONFIG_UPDATE.COM procedure
  example • 8–3

NETCONFIG_UPDATE.COM procedure (cont'd.)
running • 8–2
Network
requirement for upgrade • 5–5
restriction on upgrade • 4–2

O

Output during UETP
terminal and line printer • 9–30

P

Page file
size required for upgrade • 5–4
Password
modifying for upgrade, VAX–11/750 • 7–5
Phase controller for UETP
See UETPHAS00.EXE
Print Dialog Box
logical names • 10–8
Print formats
defining for VMS DECwindows • 10–8
Print queue
associating with a print format • 10–8
Privilege
required for UETP • 9–20
Products (VMSINSTAL)
specification for upgrade • 7–2

Q

Queues
requirement for upgrade • 5–5
QUORUM parameter
setting for concurrent upgrade • 6–2
setting for rolling upgrade • 6–3
Quota
required for UETP • 9–20

R

RMS Journaling
registering the license • 3–1
Rolling upgrade
description • 6–1
procedure • 6–2
VMS version requirement • 4–1

S

Satellites nodes
upgrade requirement • 4–3
Server startup • 10–6
Single system upgrade
procedure • 7–1 to 7–12
Source-drive
definition • 2–1, 7–1
STARTUP_P1 parameter
requirement for upgrade • 5–4
SYS$INPUT logical name • 9–29
SYS$OUTPUT logical name • 9–30
SYS$TEST logical name • 9–3, 9–9, 9–17
SYSGEN
parameter requirement for upgrade • 5–4
System
logging into for UETP • 9–1, 9–3
resource requirements for UETP • 9–1, 9–3
System disk
checking amount of free space on • 3–2
free space required for upgrade • 5–4
space needed to decompress libraries • 3–2
space requirements for UETP • 9–4
test error during UETP • 9–21, 9–22
UETP test image • 9–31
UETP test of • 9–30
using DECW$TAILOR to customize • 3–4, 8–9, F–1
using VMSTAILOR to customize • 3–3, 8–8
System hang • 7–12, 9–18, 9–26
SYSTEST account
logging into for UETP • 9–1, 9–3
privileges required for UETP • 9–20
quotas required for UETP • 9–20
SYSTEST directory
creating for UETP • 9–5
function during UETP • 9–3
SYSTEST_CLIG account
reenabling for UETP • 9–8
requirements for UETP • 9–8, 9–35
SYSUAF.DAT
restoring after upgrade • 8–1

T

Tailored system disk
  See VMSTAILOR
Tape cartridge drive
  preparing for UETP • 9–6
Target-drive
  definition • 2–1, 7–1
TCP/IP software
  use with DECwindows • 10–6
Terminal
  preparing for UETP • 9–2, 9–4, 9–6, 9–11
  simulating users for UETP • 9–31
  test of • 9–28
  UETP output • 9–30
  UETP test image • 9–31
  UETP test of • 9–30

U

UETCONT00.DAT file • 9–28
  creation of • 9–28
UETINIDEV.DAT file • 9–28, 9–29
  creation of • 9–28
  format • 9–29, 9–30
UETININET.DAT • 9–33
UETININET.DAT file • 9–33
UETINIT00.EXE image • 9–28
UETINIT01.EXE image • 9–18, 9–28
UETLOAD00.DAT • 9–31
UETNETS00.EXE • 9–33
UETP
  See User Environment Test Package
UETP$NODE_ADDRESS logical name • 9–10
UETP.COM file • 9–27
  termination of • 9–13
UETP.LOG file • 9–12, 9–17, 9–25, 9–32
UETPHAS00.EXE program • 9–27, 9–28
UETUNAS00.EXE UETP test image • 9–16
UIC
  See User Identification Code
UPDATE_CLUSTER_MEMBERS.COM
  running • 8–4
Upgrade procedure
  definition • 1–2
  Phase 2 • 7–8
  Phase 3 • 7–9
  Phase 4 • 7–10
Upgrade procedure (cont'd.)
  Phase 5 • 7–10
  Phase 6 • 7–11
  restrictions • 1–2
User disk
  preparing for UETP • 9–1, 9–5
  space requirements for UETP • 9–4
  test error during UETP • 9–21
  UETP test image • 9–31
  UETP test of • 9–30
User Environment Test Package (UETP)
  aborting execution of • 9–13
  description of • 9–1
  displaying tests as they run • 9–15
  initialization phase • 9–28
  interpreting output of • 9–14
  master command procedure • 9–27
  normal completion of • 9–12
  organization of • 9–27
  required privileges • 9–20
  required quotas • 9–20
  requirements for small disk systems • 9–9
  running all phases of • 9–2
  running individual phase of • 9–11
  running multiple passes of • 9–11, 9–17
  starting • 9–10
  typical failures reported by • 9–18
  when to run • 3–2, 8–8
User files
  processed during upgrade • 7–9
User Identification Code (UIC)
  for UETP • 9–5
User load
  defined for UETP DECnet–VAX test • 9–33
  defining for the UETP load test • 9–12
  equation used to determine for UETP load test • 9–16

V

VAXcluster environment
  building • 3–2
  preparing for UETP • 9–9
  registering the VMS license • 3–1
  test failure during UETP • 9–23
VAXcluster upgrade
  See Concurrent upgrade
  See Rolling upgrade
VAXVMSSYS.PAR parameter files
  conversion • 7–11

VMS DECwindows software
  choosing options during an installation • 2–8, 7–6
  compiling Ada software • 10–9
  components • 2–9, 7–7
  customizing • 10–1
  customizing the server • 10–3
  installation of • 2–9, 7–7
  installing with DECW$TAILOR • F–1
  logical names • 10–2
  starting • 10–1
  transports • 10–6
VMSINSTAL
  invoke for upgrade • 7–1
VMS license • 3–1
VMSTAILOR
  adding files to a system disk • 8–8
  removing files from a system disk • 3–3, 8–8
VMS/ULTRIX Connection software (UCX)
  function • 10–6
VMS version required, for upgrade • 4–1
Volume label • 2–7
Volume Shadowing
  registering the license • 3–1
VOTES parameter
  checking for rolling upgrade • 6–2
  setting for concurrent upgrade • 6–2
  upgrade requirement • 6–2

W

Workstation
  customizing • 10–3

How to Order Additional Documentation

Technical Support

If you need help deciding which documentation best meets your needs, call 800-343-4040 before placing your electronic, telephone, or direct mail order.

Electronic Orders

To place an order at the Electronic Store, dial 800-DEC-DEMO (800-332-3366) using a 1200- or 2400-baud modem. If you need assistance using the Electronic Store, call 800-DIGITAL (800-344-4825).

Telephone and Direct Mail Orders

Your Location	Call	Contact
Continental USA, Alaska, or Hawaii	800-DIGITAL	Digital Equipment Corporation P.O. Box CS2008 Nashua, New Hampshire 03061
Puerto Rico	809-754-7575	Local Digital subsidiary
Canada	800-267-6215	Digital Equipment of Canada Attn: DECdirect Operations KAO2/2 P.O. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6
International	————	Local Digital subsidiary or approved distributor
Internal[1]	————	SSB Order Processing - WMO/E15 *or* Software Supply Business Digital Equipment Corporation Westminster, Massachusetts 01473

[1]For internal orders, you must submit an Internal Software Order Form (EN-01740-07).

Reader's Comments

VMS Version 5.3
Upgrade and Installation Procedures
AA–NG61B–TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

I rate this manual's:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	☐	☐	☐	☐
Completeness (enough information)	☐	☐	☐	☐
Clarity (easy to understand)	☐	☐	☐	☐
Organization (structure of subject matter)	☐	☐	☐	☐
Figures (useful)	☐	☐	☐	☐
Examples (useful)	☐	☐	☐	☐
Index (ability to find topic)	☐	☐	☐	☐
Page layout (easy to find information)	☐	☐	☐	☐

I would like to see more/less _____

_____

_____

What I like best about this manual is _____

_____

_____

What I like least about this manual is _____

_____

_____

I found the following errors in this manual:

Page	Description
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:

_____

_____

_____

_____

I am using **Version** _____ of the software this manual describes.

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

_____ Phone _____

------ Do Not Tear - Fold Here and Tape ------

digital™

No Postage
Necessary
if Mailed
in the
United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
Corporate User Publications—Spit Brook
ZK01-3/J35 110 SPIT BROOK ROAD
NASHUA, NH 03062-9987

------ Do Not Tear - Fold Here ------

Reader's Comments

VMS Version 5.3
Upgrade and Installation Procedures
AA–NG61B–TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

I rate this manual's:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	☐	☐	☐	☐
Completeness (enough information)	☐	☐	☐	☐
Clarity (easy to understand)	☐	☐	☐	☐
Organization (structure of subject matter)	☐	☐	☐	☐
Figures (useful)	☐	☐	☐	☐
Examples (useful)	☐	☐	☐	☐
Index (ability to find topic)	☐	☐	☐	☐
Page layout (easy to find information)	☐	☐	☐	☐

I would like to see more/less _____

_____

_____

What I like best about this manual is _____

_____

_____

What I like least about this manual is _____

_____

_____

I found the following errors in this manual:

Page	Description
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:

_____

_____

_____

_____

I am using **Version** _____ of the software this manual describes.

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

_____ Phone _____

------- Do Not Tear - Fold Here and Tape -------

digital™

No Postage
Necessary
if Mailed
in the
United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
Corporate User Publications—Spit Brook
ZK01-3/J35 110 SPIT BROOK ROAD
NASHUA, NH 03062-9987

------- Do Not Tear - Fold Here -------