ULTRIX

digital

Guide to the Error Logger

Order Number: AA–ME95B–TE

ULTRIX

Guide to the Error Logger

Order Number: AA-ME95B-TE
June 1990

Product Version: ULTRIX Version 4.0 or higher

The ULTRIX operating system records information about error conditions, informational messages, application messages, and other system events in the system error logging file. This guide provides an overview of the components and operation of the error logging subsystem. It also describes the `elcsd.conf` file, the `eli` command, and the error report formatter, `uerf`.

digital equipment corporation
maynard, massachusetts

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 of DFARS 252.227-7013.

© Digital Equipment Corporation 1990,1987
All rights reserved.

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 or its affiliated companies.

The following are trademarks of Digital Equipment Corporation:

digital	DECstation	ULTRIX
	DECUS	ULTRIX Mail Connection
CDA	DECwindows	ULTRIX Worksystem Software
DDIF	DTIF	VAX
DDIS	MASSBUS	VAXstation
DEC	MicroVAX	VMS
DECnet	Q-bus	VMS/ULTRIX Connection

UNIX is a registered trademark of AT&T in the USA and other countries.

IBM is a registered trademark of International Business Machines Corporation.

MICOM is a registered trademark of MICOM Systems, Inc.

Contents

About This Manual

Audience v

Organization v

Conventions v

1 Error Logging Subsystem Overview

1.1 Overview of Error Logging Components and Relationships 1–1

1.2 Overview of the Error Logging Operation 1–2

1.2.1 Local System Crash 1–2
1.2.2 Remote System Shutdown 1–2
1.2.3 Remote System Crash 1–3

2 Kernel Error Logging Subsystem Components

2.1 Data Collection Routines 2–1

2.2 Memory Resident Error Logging Buffer 2–1

2.3 The elcsd Daemon 2–1

2.4 The elcsd.conf File 2–2

2.5 The elcsd.conf Default File Parameters 2–3

2.5.1 Status 2–3
2.5.2 Errorlog File Size 2–4
2.5.3 Errorlog Directory Path 2–4
2.5.4 Backup Errorlog Directory Path 2–4
2.5.5 Single-User Errorlog Directory Path 2–4
2.5.6 Remote Hosts Errorlog Directory Path 2–4
2.5.7 Remote Host Name To Log To 2–4
2.5.8 Hosts to Log 2–5
2.5.9 Sample Server and Client elcsd.conf Files 2–5

2.5.9.1 Server system file entries 2–5
2.5.9.2 Client system file entries 2–6

3 Error Logging Subsystem Management and Maintenance

3.0.10 Disabling Error Logging 3–1
3.0.11 Enabling Error Logging 3–1
3.0.12 Enabling Error Logging in Single-User Mode 3–1
3.0.13 Clearing the Errorlog Buffer 3–2
3.0.14 Logging Messages to the Errorlog File 3–2
3.0.15 Restart the elcsd Daemon 3–2

4 The Error Report Formatter

4.1 Generating Reports 4–1

4.1.1 Formatting Reports 4–2
4.1.2 Generating Reports from Specific Files 4–4
4.1.3 Generating Reports from Specific Host Systems 4–4
4.1.4 Reporting Errors As They Occur 4–4
4.1.5 Generating Reports in Reverse Chronological Order 4–4
4.1.6 Generating Reports in Single-User Mode 4–4
4.1.7 Generating Summary Reports 4–5

4.2 Selecting Error Types 4–5

4.2.1 Selecting Adapter Errors 4–5
4.2.2 Selecting Class Errors 4–6
4.2.3 Selecting Disk Errors 4–6
4.2.4 Selecting Mainframe Errors 4–6
4.2.5 Selecting Operating System Errors 4–6
4.2.6 Selecting Tape Errors 4–7

4.3 Selecting Specific Errors 4–7

4.3.1 Selecting Errors by Record Type 4–7
4.3.2 Selecting Errors by Sequence Number 4–9
4.3.3 Selecting Errors Using a Time Range 4–9
4.3.4 Excluding Error Types 4–10

4.4 Using uerf Options 4–10

4.5 uerf Option Reference 4–10

Examples

2-1: The elcsd.conf File with Default Entries 2–2
2-2: elcsd.conf File Entries for a Server System 2–5
2-3: elcsd.conf File Entries for a Client System 2–6
4-1: The uerf Help Display 4–2
4-2: Memory Error in Brief Format 4–3

Tables

4-1: uerf Command Options 4–11

About This Manual

The ULTRIX operating system records information about error conditions, informational messages, application messages, and other system events in the system error logging file. The ULTRIX operating system provides an error logging subsystem that records the data in a log file and an error log report formatter utility that interprets the data and generates reports.

The error logging report formatter is intended for use as a system management and maintenance tool to determine the source, frequency, and type of recurrent system and hardware activity.

Audience

The audience for this manual is anyone who manages error information on the system. While this person is usually the system manager, others, such as any DIGITAL service representative, can use the Error Logging System for analyzing error information to help identify the cause of problems in the system. This guide assumes that the reader is familiar with basic ULTRIX system functions.

Organization

Chapter 1	Error Logging Subsystem Overview Provides an overview of the components and operation of the error logging subsystem.
Chapter 2	Kernel Error Logging Subsystem Components Describes the components of the error logging subsystem that are included with the ULTRIX kernel. It also describes the `elcsd.conf` file.
Chapter 3	Error Logging Subsystem Management and Maintenance Describes the `eli` command, which allows you to manually control error logging functions.
Chapter 4	The Error Report Formatter Describes how to generate and format error reports. It also describes how to select specific error types for specific purposes.

Conventions

The following conventions are used in this manual:

`cat(1)`	Cross-references to the *ULTRIX Reference Pages* include the appropriate section number in parentheses. For example, a

	reference to `cat`(1) indicates that you can find the material on the `cat` command in Section 1 of the Reference Pages.
UPPERCASE lowercase	The ULTRIX system differentiates between lowercase and uppercase characters. Literal strings that appear in text, examples, syntax descriptions, and function definitions must be typed exactly as shown.
rlogin	In syntax descriptions and function definitions, this typeface is used to indicate terms that you must type exactly as shown.
`user input`	This bold typeface is used in interactive examples to indicate typed user input.
`system output`	This typeface is used in interactive examples to indicate system output and also in code examples and other screen displays. In text, this typeface is used to indicate the exact name of a command, option, partition, pathname, directory, or file.
#	A number sign is the default superuser prompt.
>>> CPU*nn*>>	The console subsystem prompt is two right angle brackets on RISC systems, or three right angle brackets on VAX systems. On a system with more than one central processing unit (CPU), the prompt displays two numbers: the number of the CPU, and the number of the processor slot containing the board for that CPU.
. . .	A vertical ellipsis indicates that a portion of an example that would normally be present is not shown.
RETURN	This symbol is used in examples to indicate that you must press the named key on the keyboard.

Error Logging Subsystem Overview 1

The error logging subsystem is a system management utility designed to help you and your Digital field service representative analyze stored error information. The error logging subsystem records and reports errors and other events that occur on your ULTRIX system. Together with the error report formatter, you can produce error reports to help you identify the cause of problems with your system.

This chapter describes the error logging subsystem. It covers:

- Error logging subsystem components
- Error logging subsystem operation

The components of the error logging subsystem that are included with the ULTRIX kernel are described in Chapter 2. These components include the `elcsd` daemon and the `elcsd.conf` file.

Chapter 3 describes the `eli` command, which you can use to control error logging subsystem functions manually. These functions include enabling error logging in single-user mode and clearing the kernel buffer.

The error logging subsystem also contains the ULTRIX error report formatter, `uerf`, which allows the user to generate and format several specific types of error reports. Information on `uerf` is contained in Chapter 4.

In all discussions of the error logging subsystem, the following definitions apply:

server: The system that receives error messages from other systems.

client: Any system that logs messages to the server.

1.1 Overview of Error Logging Components and Relationships

The error logging subsystem consists of:

- Data collection routines that exist in device drivers, the ULTRIX kernel, or application software
- The memory resident error logging buffer located in the ULTRIX kernel
- The `elcsd` daemon
- The `elcsd.conf` file
- The `eli` command
- The `uerf` utility

The error logging system files reside in the `/etc` directory.

1.2 Overview of the Error Logging Operation

Error log events are initiated by hardware errors, informational events, the ULTRIX kernel, or applications. Appropriate information is gathered by the applicable driver, ULTRIX kernel, or application to form an error log event that is temporarily stored in the memory resident error log buffer. The error log daemon, `elcsd`, retrieves those events and transfers them to an error log file for permanent storage.

The error logging subsystem is capable of establishing a connection to a remote system for storage of error log data. See Section 2.5 for more information.

1.2.1 Local System Crash

When the system crashes, the data in the memory resident error log buffer is saved. When the system reboots, the data is retrieved and appended to the error log file for permanent storage.

The following default entry in the `/etc/rc.local` file causes a core dump and the error log data to be saved when the system crashes:

```
/etc/savecore /usr/adm/crash        >/dev/console
```

The core dump, (`vmcore` and `vmunix`), is stored in `/usr/adm/crash`. The memory resident error log data is stored in `/usr/adm/syserr/elbuffer`. When the system is rebooted and the `elcsd` daemon started, the `elcsd` daemon will retrieve the data from the `/usr/adm/syserr/elbuffer` file and append it to the error log file for permanent storage. The daemon then removes the `/usr/adm/syserr/elbuffer` file.

You can save the memory resident error log data without saving the core dump. However, saving the core dump is recommended, when the disk space is available. If you do not have adequate disk space to save the system core dump, but want to save the memory resident error log data, you need to append the `-e` option to the savecore entry in your `rc.local` file. The following is a sample savecore entry:

```
/etc/savecore -e /usr/adm/crash     >/dev/console
```

See the *Guide to System Crash Recovery* for more information on `savecore` and system crashes.

Note

You must have a savecore entry in the `rc.local` file to save data in the kernel errorlog buffer after a crash. If you do not, you may not be able to determine why the system crashed.

1.2.2 Remote System Shutdown

If client systems are logging errors to a server and the server system is shut down for maintenance or repairs, the client system logging to the host is automatically informed of the shutdown. The client system then begins logging errors to its own errorlog file. When the server system comes back up, the client is informed that service has been restored and resumes logging error messages to the server.

After a shutdown occurs, there are two files containing client messages:

- The errorlog file on the server system
- The errorlog file on the client system

1.2.3 Remote System Crash

If client systems are logging errors to a server and the server system crashes, the client system is not informed of the crash and continues trying to log errors to the server. Because the server is down, these messages are not saved. When the server system reboots, the error messages from the client are again logged in the error log file on the server system.

If the server system is going to be down for some time, you can reconfigure the client remote system by changing its `elcsd.conf` file to log locally by changing `/etc/elcsd.conf`. Then, use the `eli` command with the `-r` option to reconfigure error logging on the server.

If a client system goes down while logging errors to the server system, the server system is not affected. When the client system comes back up, it automatically begins sending error messages to the server again.

Kernel Error Logging Subsystem Components 2

This chapter describes the components of the error logging subsystem that are included with the ULTRIX kernel.

The kernel error logging subsystem consists of:

- Data collection routines that exist in device drivers, the ULTRIX kernel, or an application
- The memory resident error log buffer located in the ULTRIX kernel
- The `elcsd` daemon
- The `elcsd.conf` file

2.1 Data Collection Routines

The data collection routines exist in device drivers, the ULTRIX kernel, or an application. These routines collect pertinent data that is formed into an error log event that will be logged.

2.2 Memory Resident Error Logging Buffer

The memory resident error log buffer is located in the ULTRIX kernel. It provides temporary storage for the error log event data.

2.3 The elcsd Daemon

The `elcsd` daemon transfers the error log data from the memory resident buffer to an error log file for permanent storage. The startup of this daemon is controlled by an entry in the `/etc/rc` file. Because the `/etc/rc` file is executed when you boot your system in multi-user mode, the `elcsd` daemon is automatically started for you. If you are operating your system in single-user mode, you can use the `eli` command to manually start the `elcsd` daemon. For more information on the `eli` command, see Chapter 3.

Here is the `/etc/rc` file entry that starts the `elcsd` daemon:

```
[ -f /etc/elcsd ] && {
   /etc/elcsd &  echo 'start errlog daemon - elcsd'   >/dev/console
}
```

If you delete this entry, the `elcsd` daemon will not transfer error messages from the kernel errorlog buffer to the errorlog file.

See the `rc`(8) command in the *ULTRIX Reference Pages* for more information on the `/etc/rc` file.

2.4 The elcsd.conf File

The `elcsd.conf` file contains the configuration information used for the `elcsd` daemon. The `elcsd.conf` file allows you to:

- Determine the names and locations of the primary, backup, and single-user error log files
- Specify the size limit for the error log files
- Control remote error logging characteristics

Any user can view the `elcsd.conf` file, but you must have superuser privileges to modify the contents of this file. You can modify the contents of the `elcsd.conf` file by using the `eli` command.

You can log error messages between systems by setting up an errorlog file on a server system to receive error messages for a client system. For example, if you have an ULTRIX system with limited disk space, you can log its error messages to a larger system. However, you must specify in the `elcsd.conf` files on both systems how you want to log errors. The client `elcsd.conf` file must specify where messages are to be sent, and the server `elcsd.conf` file must have an entry naming the client system it will accept messages from. Entries in the server `elcsd.conf` file must also specify the pathname for the client messages file.

To enable error logging between systems, you must have the following entry in the `/etc/services` file for both server and client systems:

```
elcsd   704/udp #error log daemon
```

Note

You can only log errors to another system when the systems share the same network. See the *Guide to Networking* for information on networking.

Example 2-1 shows an `elcsd.conf` file with default entries. These entries allow local logging only.

Example 2-1: The elcsd.conf File with Default Entries

```
#static char    *sccsid = "@(#)elcsd.conf    (ULTRIX)     ";
#
#   elcsd - errlog configuration file
#
{           # delimiter DON'T remove or comment out!
1           # 1-local,2-logrem,4-remlog,5-remlog+priloglocal
            # errlog file size limit num. of blocks
/usr/adm/syserr         # errlog dir. path
               # backup errlog dir. path
/              # single user errlog dir. path
/usr/adm/syserr         # log remote hosts errlog dir. path
            # remote hostname to log to
}           # delimiter DON'T remove or comment out!
#   hosts to log :S - separate file or :R - remotes file (together)
#remote1:S      - (example) log errors from remote1 into separate file
```

2.5 The elcsd.conf Default File Parameters

The `elcsd.conf` file default specifications are as follows:

- Log this system's error log data locally
- Use `/usr/adm/syserr` as the primary and backup error logging directory path
- Use `/` as the single-user error logging directory path
- Use `/usr/adm/syserr` as the remote hosts error logging directory path
- No systems are logging remotely to this system

See Chapter 3 for more information on using the `eli` command to modify the `elcsd.conf` file.

2.5.1 Status

The first parameter in the `elcsd.conf` file is the status line, which indicates where error messages are logged.

The possible status conditions and their meanings are:

1-local	Log local messages here.
2-logrem	Log messages from remote systems here.
3	Log local and remote messages here.
4-remlog	Log messages from this system to a remote system.
5-remlog+priloglocal	Log messages from this system to a remote system, and log server as well as high-priority messages here.

When you log error messages between systems, you must decide how to log the different priority levels of errors, based on the amount of disk space you have available.

There are three priority levels: severe, high, and low. A severe error means the system is going down. High-priority errors are errors such as recoverable machine checks and hard errors. Low-priority errors include soft errors, restarts, and CRDs (corrected read data).

The default status condition is 1 (local). Errors that occur on the local system are logged on the local system.

Enter 2 (logrem) on the status line if you are a server system and you want to log messages from a client system.

Enter 3 on the status line to log server (local) and client (remote) messages.

Enter 4 (remlog) on the status line if you are a client system and you want to log messages to a server.

Enter 5 (remlog+priloglocal) on the status line if you have sufficient disk space and, in addition to logging all messages to a server, want to log severe and high-priority messages on your client system.

No matter what status entry you specify, severe kernel messages are sent to the local system console.

2.5.2 Errorlog File Size

The errorlog file size parameter defines the maximum size of every errorlog file on the system. If you leave the errorlog file size parameter blank, the system will notify you when the file system is 98% full. Otherwise, specify the maximum number of blocks (for example, 512 bytes) that you want for each errorlog file.

When you do not specify the size of the errorlog file, a message is written to the errorlog status file, `/usr/adm/elcsdlog`, when the file system is 98% full. At this point, you should compress or back up and remove the current errorlog file before the system stops logging messages to disk. A message is also written to the system log. See the `syslog`(8) command in the *ULTRIX Reference Pages.*

2.5.3 Errorlog Directory Path

To start an errorlog file, you must specify a directory path in the `elcsd.conf` file.

The errorlog directory path specifies the main errorlog file. The default path is `/usr/adm/syserr`. You can, however, direct error messages to a different directory. If you do, first verify that the alternate directory exists and then specify this alternate when you invoke the `uerf` command.

2.5.4 Backup Errorlog Directory Path

The backup errorlog directory path is the file the system writes to when the main errorlog file becomes full or when there is an error and the system cannot access the main file. There is no default path. Consequently, you should specify a backup errorlog directory that is different from the main errorlog directory.

2.5.5 Single-User Errorlog Directory Path

The single-user errorlog file is `/syserr.`*hostname*. The root (`/`) directory is the default location for this file. You can direct single-user error messages to a different directory, but you must be sure that the directory you specify is mounted during single-user mode. When the system makes the transition from single-user mode to multiuser mode, errors logged in `/syserr.`*hostname* are appended to `syserr.`*hostname* in the multiuser errorlog directory path and the single-user errorlog file is removed.

2.5.6 Remote Hosts Errorlog Directory Path

The remote hosts errorlog directory path is the directory for the errorlog file containing messages logged from remote, or client systems. The default path is `/usr/adm/syserr`. It is a good idea to use the same path name that you used for the main errorlog directory.

2.5.7 Remote Host Name To Log To

This parameter specifies the name of the remote (server) system that you are logging to. This entry belongs in the client `elcsd.conf` file. There is no default remote host name.

2.5.8 Hosts to Log

The hosts to log parameter specifies the name of the remote (client) system that you are logging to, and specifies how to log the client's messages. The parameter takes two arguments:

- The argument, S, sets up a separate errorlog file, `syserr.`*hostname*, for each client system.
- The default argument, R, logs error messages from all client systems to the `syserr.remotes` file.

2.5.9 Sample Server and Client elcsd.conf Files

The following examples show and describe the contents of a server `elcsd.conf` file (see Example 2-2) and a client `elcsd.conf` file (see Example 2-3). These examples are based on the following case:

Assume that you want to log error messages to a VAX server, `piano`, from two MicroVAX clients, `flute` and `violin`. You have decided to set up individual errorlog files on the server for each remote client. In addition, because your client systems have adequate disk space, you decide to log severe and high-priority error messages on each client system.

2.5.9.1 Server system file entries – To set up the required server files, you edit the server's `elcsd.conf` file as shown in Example 2-2.

Example 2-2: elcsd.conf File Entries for a Server System

```
#static char *sccsid = "@(#)elcsd.conf    (ULRIX) ";
#
#          elcsd - errlog configuration file
#
{     # delimiter DON'T remove or comment out!
3     # status 1-local,2-logrem,4-remlog,5-remlog+priloglocal
      # errlog file size limit num. of blocks
/usr/adm/syserr # errlog dir. path
                          # backup errlog dir. path
/                         # single user errlog dir. path
/usr/adm/syserr # log remote hosts errlog dir. path
      # remote hostname to log to
}     # delimiter DON'T remove or comment out!
#   hosts to log :S - separate file or :R - remotes file (together)
flute:S
violin:S
```

If you examine the contents of this file, you will notice the following entries:

- The error logging status, 3, indicates that both server and client messages are to be logged to the server, `piano`.
- The errorlog directory path default, `/usr/adm/syserr`, is specified as the main errorlog directory.
- The remote hosts errorlog directory path default, `/usr/adm/syserr`, is specified as the directory for error messages logged from remote (client) systems.

- Both `flute` and `violin` are listed as the hosts to log. Consequently, the VAX server will log messages for each of these client systems. Including the `S` argument with each host name (flute:S, for example) specifies that each client will have its own errorlog file on the server. These individual files will be subordinate to the remote hosts errorlog directory, `/usr/adm/syserr`. Therefore, error messages for `flute` will reside on the VAX server at `/usr/adm/syserr/syserr.flute`. Similarly, error messages for `violin` will reside on the VAX server at `/usr/adm/syserr/syserr.violin`.

2.5.9.2 Client system file entries – To set up the required client files, you edit the `elcsd.conf` file for both `flute` and `violin`. The following example shows the client file entries.

Example 2-3: elcsd.conf File Entries for a Client System

```
#static char    *sccsid = "@(#)elcsd.conf     (ULTRIX)  ";
#
#          elcsd - errlog configuration file
#
{     # delimiter DON'T remove or comment out!
5     # status 1-local,2-logrem,4-remlog,5-remlog+priloglocal
      # errlog file size limit num. of blocks
/usr/adm/syserr # errlog dir. path
                          # backup errlog dir. path
/                         # single user errlog dir. path
/usr/adm/syserr # log remote hosts errlog dir. path
piano           # remote hostname to log to
}               # delimiter DON'T remove or comment out!
#   hosts to log :S - separate file or :R - remotes file (together)
```

If you examine the contents of this file, you will notice the following entries:

- The error logging status, `5`, specifies that all error messages are logged to the server, and that severe and high-priority error messages are logged to the client as well.
- The errorlog directory path default, `/usr/adm/syserr`, is specified as the main errorlog directory.
- The remote hosts errorlog directory path default, `/usr/adm/syserr`, is specified as the directory for error messages logged from remote (server) systems.
- The server, `piano`, is listed as the remote hostname to log to. Consequently, the client will log messages to this server system.

Error Logging Subsystem Management and Maintenance 3

The `eli` command lets you manually control these error logging functions:

- Enable and disable error logging
- Clear the errorlog buffer
- Log messages to the errorlog file
- Restart the `elcsd` daemon

The following sections describe how to use `eli` to perform these functions.

Refer to `eli`(8) in the *ULTRIX Reference Pages* for more information on the command options.

3.0.1 Disabling Error Logging

To disable error logging to an errorlog file, type:

```
# /etc/eli -d
```

You can use this option when your file system is full, while you free up more disk space. However, when you use this option, the `elcsd` daemon stops logging error messages. Messages queue up in the kernel buffer until the buffer becomes full. These messages will be logged to disk when the error log daemon is restarted.

3.0.2 Enabling Error Logging

To re-enable error logging in multiuser mode, type one of the following command lines:

```
# /etc/eli -e
```

```
# /etc/eli -f -e
```

When you use the `-f` option, the system suppresses the prompts you usually see when you select the `-e` option. The `elcsd` daemon creates a new errorlog file. The default pathname is `/usr/adm/syserr/syserr.`*hostname*.

3.0.3 Enabling Error Logging in Single-User Mode

Error messages can be logged during single-user mode and multiuser mode. In single-user mode, however, you must start up the `elcsd` daemon manually to log error messages to the errorlog file.

Before starting the `elcsd` daemon in single-user mode, run `fsck` on the file system you will be logging to check for file system inconsistencies. (The root file system is the default file system for error logging in single-user mode.) Then, invoke the `eli` command by typing:

```
# eli -s
```

The -s option enables error logging in single-user mode. Error messages are transferred from the kernel errorlog buffer to the file specified in the elcsd.conf file for messages generated during single-user mode.

The root (/) directory is the default location for the single-user errorlog file. When the system makes the transition to multiuser mode, by default, it appends the messages in /syserr.*hostname* to the end of the multiuser errorlog file, syserr.*hostname* in /usr/adm/syserr. If you do not want to save error messages generated during single-user mode, remove the single-user errorlog file before you make the transition to multiuser mode. To change the single-user directory path or any other parameters in the elcsd.conf file, see Section 2.5.

3.0.4 Clearing the Errorlog Buffer

To clear the kernel errorlog buffer, type:

```
# /etc/eli -i
```

This command line initializes the kernel errorlog buffer. The messages in the kernel errorlog buffer are cleared without being logged to disk.

If a problem occurs anywhere in the system that stops the elcsd daemon from running, an error message informing you that error messages are not being logged to an error log file appears on the console every 15 minutes. To stop this message, type:

```
# /etc/eli -q
```

To re-enable logging this missed-error message to the console, type:

```
# /etc/eli -w
```

3.0.5 Logging Messages to the Errorlog File

You can log a 1-line message to the errorlog file with this command line:

```
# /etc/eli -l
```

The system prompts you to enter the message. You can also log a message that is contained in a file. For example, using the file myfile, the command line entry is;

```
# eli -f -l < myfile > /dev/null
```

This example logs a message, up to and including the first newline, from the file named myfile. The example also directs the eli prompt to /dev/null.

3.0.6 Restart the elcsd Daemon

If you change any entries in the elcsd.conf file, you must restart the elcsd daemon to make the new parameters take effect. Type:

```
# eli -r
```

The Error Report Formatter 4

The ULTRIX error report formatter, `uerf`, reports errors and events that occur on your ULTRIX system. The `uerf` utility accesses the error messages and events stored in the errorlog file, translates them from binary code to ASCII, and sends them to the output device you specify.

The `uerf` utility uses three data files. The `uerf.bin` file is the event information data base, `uerf.hlp` is the help file, and `uerf.err` is the error message file.

The `uerf` utility searches for the `uerf` data files in:

- The directory specified in a full pathname with the `uerf` command
- The `/etc` directory
- The directories specified in your shell PATH environment variable

The `uerf` utility has options that let you generate various types of reports on specified errors. For example, you can produce a report containing all the error messages in the errorlog file, or you can produce a report, using options, with only a few error types. In relation to these options, this chapter provides information on:

- Generating reports
- Selecting error types
- Selecting specific errors

This chapter also contains a quick reference for using `uerf`. You do not need superuser privileges to use the `uerf` command.

4.1 Generating Reports

The `uerf` utility produces reports based on the error messages stored in an errorlog file. Using `uerf` with its options, you can produce error reports to a terminal screen, to a file, or to a printer. For example, to generate an error report and display it on your terminal one screenful at a time, type:

```
# /etc/uerf | more
```

By looking at the types and number of errors and events, you can tell how reliable the system is. If a report shows a large number of errors for a particular device, you can tell that there is a problem before the device fails. Furthermore, if a failure does occur, the error report provides information on the events that led up to the failure.

The error logging system records and reports the following errors and events:

- Errors – Device errors, device timeouts, machine checks, bus errors, and memory errors, such as hard or soft error correcting code (ECC) errors
- Events – System startup and shutdown messages, system failures or panics, and informational messages

To see what options are available with `uerf`, use the help option:

```
# /etc/uerf -h
```

This displays a list of available options and their meanings. Example 4-1 shows the `uerf` help display.

The rest of this section explains the `uerf` options. These options enable you to:

- Format reports
- Generate reports from specific files
- Generate reports from specific host systems
- Report on errors as they occur
- Generate reports in reverse chronological order
- Generate reports in single-user mode
- Generate summary reports.

For a detailed list of options in quick reference format, see Section 4.5 or `uerf`(8) in the *ULTRIX Reference Pages.*

Example 4-1: The uerf Help Display

```
                        U E R F   H E L P
 uerf<cr>             - process all events in the default input file
PROCESSING OPTIONS
  -h                  - display this help screen.
  -f input_filename   - process events from the specified input file.
  -n                  - process events from kernel errorlog buffer immediately.
  -H ascii_host_name - process events for the specified hosts.
  -R                  - process events from the errorlog file in reverse order.
  -S                  - display a summary of selected events.
  -t time_range       - process events in the specified time range.
  -o output_format    - format the output in brief, full or terse.
  -x                  - exclude selected events (below) from output.
  -Z                  - display the entry in hex format
SELECTION OPTIONS - Events may be selected as follows.
  -A adapter_list          -D disk_devices          -M mainframe_type
  -O os_specific           -T tape_devices          -c class_of_events
  -e error_types           -r record type           -s sequence number
            ***** For a detailed list of options see uerf(8) *****
```

4.1.1 Formatting Reports

Use the `-o` option to format `uerf` error reports. This option enables you to format reports in three ways:

brief — Reports event information in a short format.

full — Reports all available information for each error.

terse — Reports binary event information and displays register values and other ASCII messages in a condensed format.

This option requires a parameter. If you do not select the `-o` option, the default output format is brief.

Most error types produce more information with the `-o full` qualifier. There are a few, such as panic messages and other ASCII messages, which do not.

The brief output format is shown in Example 4-2, for this command line:

```
# /etc/uerf -M mem
```

Example 4-2: Memory Error in Brief Format

```
****************************** ENTRY  2 ******************************
----- EVENT INFORMATION SEGMENT -----
EVENT CLASS                              ERROR EVENT
OS EVENT TYPE                   101.     MEMORY ERROR
SEQUENCE NUMBER                   3.
OPERATING SYSTEM                         ULTRIX 32
OCCURRED/LOGGED ON                       Thu May 14 18:14:46 1987 EDT
OCCURRED ON SYSTEM                       guitar
SYSTEM ID                 x01845106
PROCESSOR TYPE                           KA780
----- UNIT INFORMATION -----
UNIT CLASS                               MEMORY
UNIT TYPE                                MS780E
ERROR SYNDROME                           MEMORY CRD ERROR
----- 780-E CSR REGS -----
CSRA                      x00101E6C      INTERLEAVE:  INTERNAL 2-WAY
                                         RAM:  64K
                                         ADAPTER CODE:  x3
                                         MEMORY SIZE:  15.
                                         ERROR SUMMARY
CSRB                      x00001000      DIAG ECC BITS:  x0
                                         MEM HAS VALID DATA
                                         START ADDR:  x0
CSRC                      x148F020E      ERR SYNDR/CHK BITS:  xE
                                         CRD
                                         ERROR ADDR:  x91E0
                                         ERROR LOG REQUEST
CSRD                      x080AD002      ERR SYNDR/CHK BITS:  x2
                                         ERROR ADDR:  x1015A
```

To show the full output format for this report, which displays all memory-related mainframe errors, type:

```
# /etc/uerf -o full -M mem
```

In addition to the information shown in Example 4-2, the full output format produces this information:

```
----- ADDITIONAL MEMORY INFO -----
CNTRLR NO                         1.
NO. ERRS ON THIS ADD              1.
```

To produce the terse output format for all memory-related mainframe errors, type:

```
# /etc/uerf -o terse -M mem
```

This report gives you binary event information, for the events you specify. The terse memory-related mainframe errors report looks like this:

```
MEMORY ERROR:                KA780
MEMORY CRD ERROR            MS780E
CSRA                      x00101E6C
CSRB                      x00001000
CSRC                      x00000000
```

```
CSRD                          x1A0AD275
CNTRLR NO                            1.
NO. ERRS ON THIS ADD                 1.
```

4.1.2 Generating Reports from Specific Files

The file name (`-f`) option of `uerf` selects errors from the errorlog file you specify, for example:

```
# /etc/uerf -f /usr/adm/syserr/syserr.guitar.old
```

Use this option to look at old or backup errorlog files, rather than at the default file, `/usr/adm/syserr/syserr.`*hostname*. You can also use this option to access the single-user errorlog file, `/syserr.`*hostname*, in the root directory. Be sure to specify the full path name for the file. Do not use the `-n` option with the `-f` option. You must specify a parameter with the `-f` option.

4.1.3 Generating Reports from Specific Host Systems

The host (`-H`) option of `uerf` selects errors for a specific host system from the `/usr/adm/syserr/syserr.`*hostname* or `/usr/adm/syserr/syserr.remotes` file. You must specify a host name as a parameter. This option is useful when you are logging errors from several remote systems to the errorlog file, `/syserr.remotes`, on the host. See Chapter 2 for information on remote logging.

4.1.4 Reporting Errors As They Occur

The now (`-n`) option of `uerf` reports errors to the terminal as they occur, before they are logged to the errorlog buffer. Here is the command line:

```
# /etc/uerf -n
```

You can use this option when you run disk or tape exercisers. Do not use the `-f` option with the `-n` option.

4.1.5 Generating Reports in Reverse Chronological Order

To generate reports in reverse chronological order, use the `-R` option of `uerf`. As shown in the following example, you can select `-R` to start at the most current error event logged, while selecting only certain record types (using `-r option`):

```
# /etc/uerf -R -r 300
```

In response to this command, the `uerf` program produces a report that lists all startup messages, beginning with the most recent.

4.1.6 Generating Reports in Single-User Mode

You can run `uerf` in single-user mode. However, if the errorlog file and the `uerf` data files, `uerf.bin`, `uerf.hlp`, and `uerf.err`, are not located in the root directory, you must first mount the file systems on which they reside.

In addition, the line printer spooler is not operational during single-user mode. Therefore, to print an error report to a line printer while in single-user mode, you must direct the output to a printer special file, such as the following example using *piano* as the hostname:

```
# /etc/uerf -f /syserr.piano > /dev/lp
```

4.1.7 Generating Summary Reports

The summary (`-S`) option of `uerf` produces a summary of the errorlog file or a summary of those records that you select. All `uerf` selection options support summaries. The default format for summary output is terse. However, you can generate summary output in full or brief format by including the output (`-o`) option and specifying the desired format.

The following example shows the command and options that generate a terse summary of all errors recorded for the day in the errorlog file:

```
# /etc/uerf -t s:00 -S
```

To generate a full summary of the same information, type:

```
# /etc/uerf -t s:00 -S -o full
```

4.2 Selecting Error Types

The `uerf` utility allows you to generate specific error reports for the following:

- Adapter errors (`-A`)
- Class errors (`-c`)
- Disk errors (`-D`)
- Mainframe processor errors (`-M`)
- Operating system errors (`-O`)
- Tape errors (`-T`)

For each of these error types, `uerf` has a corresponding option. Each of the options has report parameters that you can specify to generate various types of reports.

The following sections explain how to select each of these error types.

4.2.1 Selecting Adapter Errors

The `-A` option selects errors for the adapter parameters that you specify. The adapter parameters the error report formatter supports are:

aie	BVP-type controller
aio	BVP-type controller
bla	BI LESI adapter
bua	BI UNIBUS adapter
nmi	Nautilus memory interconnect
uba	VAX UNIBUS adapter

The following example shows how to generate a report for *uba* (VAX UNIBUS) errors that have been logged in the errorlog file:

```
# /etc/uerf -A uba
```

To generate a report for more than one parameter, use the comma (,) as a delimiter. For example, the following command generates a report for all `uba` and `nmi` adapter errors:

```
# /etc/uerf -A uba,nmi
```

4.2.2 Selecting Class Errors

The `-c` option selects errors for the class event that you specify. The class parameters that the error report formatter supports are:

err	Hardware-detected and software-detected error conditions
oper	Operational events, such as device status or error messages, time changes, and system startup and shutdown messages.
maint	Events that occur during system maintenance, such as running the on-line functional exercisers.

4.2.3 Selecting Disk Errors

The `-D` option selects errors for the MSCP and SCSI disk parameters you specify. See the `ra`(4) command in the *ULTRIX Reference Pages* for the MSCP disk types that the error report formatter supports.

4.2.4 Selecting Mainframe Errors

The `-M` option selects the following types of processor errors for the systems for which you are logging errors:

cpu	CPU-related errors, such as machine checks
mem	Memory-related errors, such as single-bit CRD (corrected read data) and double-bit uncorrectable errors

When you do not specify any parameters, all mainframe errors are reported.

The following command produces a report containing CPU-related mainframe errors for the system:

```
# /etc/uerf -M cpu
```

4.2.5 Selecting Operating System Errors

The `-O` option selects errors for the operating system parameters you specify. When you do not specify any parameters, all operating system events are reported. The operating system parameters that the error report formatter supports are:

aef	Arithmetic exception faults
ast	Asynchronous trap exception faults
bpt	Breakpoint instruction faults
cmp	Compatibility mode faults
pag	Page faults
pif	Privileged instruction faults
pro	Protection faults
ptf	Page table faults
raf	Reserved address faults
rof	Reserved operand faults
scf	System call exception faults
seg	Segmentation faults
tra	Trace exception faults
xfc	Xfc instruction faults

4.2.6 Selecting Tape Errors

The `-T` option selects errors for the TMSCP and SCSI tape parameters that you specify. See the `tms`(4) command in the *ULTRIX Reference Pages* for the TMSCP tape types that the error report formatter supports.

4.3 Selecting Specific Errors

Some `uerf` options let you select the following errors specifically:

- Errors specified by record type
- Errors specified by sequence number
- Errors within a specified time range
- Errors except the ones that you specifically exclude

The following subsections describe `uerf` options that let you select or exclude specific errors.

4.3.1 Selecting Errors by Record Type

The `-r` option selects errors for specific record types. You must specify a parameter that defines which record type you want to select. Use this option to access errors, such as ASCII messages, which are not accessed by other `uerf` options. The `-r` option also offers an alternate way to specify some error events, such as disk and tape errors.

Devices, such as UNIBUS communications devices, log to the errorlog file in an ASCII rather than binary message format.

Note

It is possible that the text of an ASCII error message, when reported in the brief or the full output format, will generate more than one error message. If another error occurs and is reported at approximately the same time, these ASCII text messages may not print out consecutively. You can use the terse output format to see such messages as one unit.

The record parameters supported by the error report formatter are:

Hardware-Detected Errors

100 Machine check
101 Memory corrected read data/read data substitute (crd/rds)
102 Disk errors
103 Tape errors
104 Device controller errors
105 Adapter errors
106 Bus errors
107 Stray interrupts
108 Asynchronous write errors
109 Exceptions/faults
112 Stack dump
113 ka650 error and status registers
114 6200 vector 60
115 6200 vector 54
116 ka420 error and status registers (VAXstation 3100)
117 kn01 error and status registers (DECstation R3100)
118 6400 vector 60
119 6400 vector 54
120 Mbus errors
121 ka60 error and status registers
130 Generic error and status registers

Software-Detected Errors

200 Panics (bug checks)
201 ci ppd info
202 scs events

Informational ASCII Messages

250 Informational
251 8600/8650 snapshot taken

Operational Messages

300 Startup
301 Shutdown
310 Time change
350 Diagnostic information
351 Repair information

The following example produces all system startup messages, including hardware devices configured and their CSR (control status register) addresses:

```
# /etc/uerf -r 300
```

The following example outputs all of the operational messages – startup, shutdown, time change, and diagnostic – within the record option:

```
# /etc/uerf -r 300-350
```

4.3.2 Selecting Errors by Sequence Number

The `-s` option selects specific errors from the errorlog file. The selection corresponds to the sequence number given as a parameter to the –s option. You must specify the sequence number. A sequence number is assigned to an event when it is logged to the errorlog file. You can use this option to output specific errors after viewing the errorlog file at your terminal.

Note

Sequence numbers are restarted when the system is rebooted. If the errorlog file contains error messages from before and after a reboot, there may be duplicate sequence numbers in the file.

4.3.3 Selecting Errors Using a Time Range

The `-t` option selects errors for the time period you specify. The format for the `-t` option is:

`/etc/uerf -t` *s:dd-mmm-yyyy,hh:mm:ss e:dd-mmm-yyyy,hh:mm:ss*

The parameters for the `-t` option are:

s	Specifies the start date and time
e	Specifies the end date and time
dd	Day
mmm	Month
yyyy	Year
hh	Hour
mm	Minute
ss	Second

You must specify a start date or time or an end date or time when you use this option. However, you can abbreviate the format using the following default parameters:

- When none is specified, the current date is the default date.
- The default start time is 00:00:00.
- The default end time is 23:59:59.

When you do not use this option, the `uerf` command processes the entire errorlog file.

The following example shows how to produce an error report containing all errors for the 24-hour period of October 23, 1988:

```
# /etc/uerf -t s:23-oct-1988,00:00:00 e:23-oct-1988,23:59:59
```

The following command produces an error report from the beginning of the errorlog file until February 29 of the current year; the default end time is 23:59:59:

```
# /etc/uerf -t e:29-feb
```

4.3.4 Excluding Error Types

The `-x` option excludes the error types that you specify from the error report. You cannot exclude the `-f` option, the `-h` option, the `-H` option, the `-n` option, the `-o` option, the `-t` option, or the `-R` option. The following command reports all errors except disk errors and operating system events:

```
# /etc/uerf -O -x -o full -D
```

4.4 Using uerf Options

This section provides some notes and examples of `uerf` use. See Table 4-1 for a quick reference of the command options.

- Be sure to type options in the correct case, because the `uerf` command has both uppercase and lowercase options. For example, `-o` specifies output format, while `-O` specifies operating system events.
- Options can be used together. For example:

  ```
  # /etc/uerf -A uba,nmi -f syserr.guitar.old -TH guitar
  ```

  The preceding command produces an error report from the file `syserr.guitar.old` for the system `guitar`. The report contains adapter errors for the VAX UNIBUS adapter and the Nautilus memory interconnect, as well as TMSCP (Tape Mass Storage Control Protocol) device errors of all types.

  In the following example, the time and output options produce error messages for the current day in a terse format at your terminal:

  ```
  # /etc/uerf -t s:00 -o terse
  ```

4.5 uerf Option Reference

Table 4-1 lists each `uerf` option alphabetically and provides brief descriptions and examples.

Table 4-1: uerf Command Options

Option	Usage	Examples
-A adapters	Select adapter and device controller errors.	/etc/uerf -A bua,nmi
-c classes	Select classes of errors.	/etc/uerf -c err
-D disks	Select mscp disk devices and SCSI disk devices.	/etc/uerf -D rx /etc/uerf -D ra60,rd31 /etc/uerf -D rz /etc/uerf -D rz23,rz24
-f filename	Specify the file from which error messages are read. Use full pathname. Do not use with the -n option.	/etc/uerf -f /usr/adm/syserr/syserr.old
-h	Display help message. Do not use with another option.	/etc/uerf -h
-H hostname	Select errors for specified system. Use for remote logging to syserr.remotes.	/etc/uerf -f /usr/adm/syserr/syserr.remotes -H guitar
-M mainframe	Specify processor error types.	/etc/uerf -M mem /etc/uerf -M cpu,mem
-n	Display errors as they occur. Do not use with -f option.	/etc/uerf -n -D
-o output	Select output format for report. Default is brief format.	/etc/uerf -o brief /etc/uerf -o full /etc/uerf -o terse
-O operating system	Select operating system errors.	/etc/uerf -O /etc/uerf -O raf,ptf,ast
-R reverse chronological order	Select records in reverse chronological order.	/etc/uerf -R /etc/uerf -A bua -R /etc/uerf -R -O
-r records	Select errors for specified record codes.	/etc/uerf -r 300,301 /etc/uerf -r 100-109
-s sequence numbers	Select errors for specified sequence numbers.	/etc/uerf -s 750-800

Table 4-1: (continued)

Option	Usage	Examples
–S	Produce summary report.	/etc/uerf -S -o brief
–t time	Select errors within specified time range.	/etc/uerf –t s:13-apr-1986 /etc/uerf –t s:13-apr,18:30
–T tapes	Select errors for tmscp tape types and SCSI tape devices.	/etc/uerf –T /etc/uerf –T tu81 /etc/uerf –T tz /etc/uerf –T tz31
–x	Exclude specified options. You cannot exclude –f, –H, –h, –n, –o, –t, or –R options.	/etc/uerf –A –x –D
–Z	Display entry in hexadecimal format.	/etc/uerf –Z

Index

C

client
defined, 1–1
components of
error logger, 1–1

D

default
elcsd.conf file specifications, 2–3

E

elcsd daemon
rc file entry, 2–1
elcsd.conf file
changing entries, 3–2
for client system, 2–6e
for server system, 2–5e
parameters, 2–3 to 2–5
reconfiguring, 2–3 to 2–5
specifying client system, 2–5
specifying server system, 2–4
with default entries, 2–2e
eli command
options, 3–1 to 3–2
error log file
specifying size limit for, 2–2
error logger
components, 1–1 to 1–3
controlling manually, 3–1 to 3–2
defined, 1–1
description, 2–1 to 2–5
disabling, 3–1
enabling in multi-user mode, 3–1
error logger (cont.)
enabling in single-user mode, 3–1
error message types, 2–3
errors reported, 4–1
events reported, 4–1
overview, 1–1 to 1–3
remote system crash and, 1–3
remote system shutdown and, 1–2
error logging subsystem
maintenance, 3–1
management, 3–1
error report
ASCII messages and, 4–7n
brief format, 43e
directing to terminal, 4–1
displaying as errors occur, 4–4
excluding error types, 4–10
for adapter errors, 4–5
for class errors, 4–6
for disk errors, 4–6
for error type, 4–5 to 4–7
for mainframe errors, 4–6
for operating system, 4–6
for record type, 4–7
for sequence number, 4–9
for specific errors, 4–7 to 4–10
for tape errors, 4–7
for time period, 4–9
formatting, 4–2 to 4–4
from specific file, 4–4
from specific systems, 4–4
full format, 4–3e
generating, 4–1 to 4–4
generating in reverse order, 4–4

error report (cont.)
generating in single-user mode, 4–4
specifying brief format, 4–3
specifying full format, 4–3
specifying terse format, 4–3
terse format, 4–2
error report formatter
See uerf command
errorlog buffer
clearing, 3–2
errorlog file
logging message to, 3–2
remote system directory path, 2–4
remote systems and, 2–3
single-user directory path, 2–4
specifying backup, 2–4
specifying directory path, 2–4
specifying size, 2–4
event file
See errorlog file

K

kernel error logging
components of, 2–1

O

overview of
error logging components, 1–1

R

remote error logging, 2–2

S

server
defined, 1–1

U

uerf command
data files, 4–1
help display, 4–2e
help option, 4–2
option list, 4–11t to 4–12t
options, 4–10 to 4–12
record parameters supported, 4–7
superuser privileges, 4–1
using, 4–1 to 4–12

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 800-234-1998 using a 1200- or 2400-baud modem from anywhere in the USA, Canada, or Puerto Rico. 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*	————	SSB Order Processing - WMO/E15 *or* Software Supply Business Digital Equipment Corporation Westminster, Massachusetts 01473

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

Reader's Comments

ULTRIX
Guide to the Error Logger
AA-ME95B-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.

Please rate this manual:	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)	☐	☐	☐	☐

What would you like to see more/less of? _____

What do you like best about this manual? _____

What do you like least about this manual? _____

Please list errors you have found in this manual:

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

Additional comments or suggestions to improve this manual:

What version of the software described by this manual are you using? _____

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

_____ Email _____ Phone _____

Do Not Tear – Fold Here and Tape

digital™

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST–CLASS MAIL PERMIT NO. 33 MAYNARD MA

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
OPEN SOFTWARE PUBLICATIONS MANAGER
ZKO3–2/Z04
110 SPIT BROOK ROAD
NASHUA NH 03062–9987

Do Not Tear – Fold Here

Cut
Along
Dotted
Line

Reader's Comments

ULTRIX
Guide to the Error Logger
AA-ME95B-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.

Please rate this manual:	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)	☐	☐	☐	☐

What would you like to see more/less of? _____

_____

_____

What do you like best about this manual? _____

_____

_____

What do you like least about this manual? _____

_____

_____

Please list errors you have found in this manual:

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

Additional comments or suggestions to improve this manual:

_____

_____

_____

_____

What version of the software described by this manual are you using? _____

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

_____ Email _____ Phone _____

Do Not Tear – Fold Here and Tape

digital™

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST–CLASS MAIL PERMIT NO. 33 MAYNARD MA

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
OPEN SOFTWARE PUBLICATIONS MANAGER
ZKO3–2/Z04
110 SPIT BROOK ROAD
NASHUA NH 03062–9987

Do Not Tear – Fold Here

Cut
Along
Dotted
Line