Tru64 UNIX DECevent Translation and Reporting Utility User's Guide

This guide is intended for users of the translation and reporting features of the DECevent event management utility on Tru64 UNIX® operating systems.

Order Number: AA-RH97A-TE

Date: July 1999

Product Version: Tru64 UNIX V5.0 or higher

Software Version: DECevent V2.9 or higher

Compaq Computer Corporation
Houston, Texas

This document was prepared using FrameMaker, Version 5.5.6.

Contents

Chapter 4 Customizing Your Environment

Chapter 5 The DECevent dia Command Verb

Chapter 6 The ls Commands

Chapter 7 The shw Commands

Chapter 8 The -int Flag

Appendix A DECevent Directories and Files

Appendix B System Settings

Appendix C Bit-To-Text Quick Reference

Appendix D DECevent Selection Keywords for Bit-To-Text Translation

List of Tables and Examples

Preface

This manual contains all DECevent command features related to the translation and reporting of events on Tru64 UNIX systems.

Command Symbols and Documentation Conventions

Various symbols are used in command expressions to indicate what information is optional and when a value or parameter is required. Symbols such as brackets [ ] and parenthesis () are used extensively. These symbols are used only for informational purposes and should never be typed in the command line.

In general, the command expressions use the symbols shown in the following table.

Symbols	Purpose
{ }	In format command descriptions, braces indicate required elements. You must include one of the elements.
()	In format descriptions, parentheses indicate that if you choose more than one option, you must enclose the choices in parentheses.
[ ]	In format descriptions, brackets indicate that whatever is enclosed within the brackets is optional; you can select one, none, or all of the choices. (Brackets are not optional, however, in the syntax of a directory name in a file specification or in the syntax of a substring specification in an assignment statement.)
[...]	Square brackets containing a space and three ellipses indicate a list of optional values separated by spaces.
. . .	Vertical ellipsis points indicate the omission of information from an example or command format. The information has been omitted because it is not critical to the topic being discussed.
italic type	Italic type emphasizes important information and indicates variables, complete titles of manuals, and parameters for system information.

Ctrl/x	Hold down the key labeled Ctrl (Control) and the specified key simultaneously (such as Ctrl/Z).

Readers' Comments

Compaq welcomes any comments and suggestions you have on this and other Tru64 UNIX manuals. You can send your comments in the following ways:

- Fax: 603-881-0120 Attn: UBPG Publications, ZK03-3/Y32
- Internet electronic mail: readers_comment@zk3.dec.com
- A Reader's Comment form is located on line at /usr/doc/readers_comment.txt
- Mail:

  Compaq Computer Corporation
  UBPG Publications Manager
  ZK03-3/Y32
  110 Spit Brook Road
  Nashua, NH 03062-9987

- A Reader's Comments form is located in the back of each printed manual. The form is postage paid, if mailed in the United States.

  Include the following information along with your comments:

  – The full title of the book and the order number. (The order number is printed on the title page of this book and on its back cover.)

  – The section numbers and page numbers of the information on which you are commenting.

  – The version of Tru64 UNIX that you are using (for example, Tru64 UNIX Version 4.0).

  – If known, the type of processor that is running the Tru64 UNIX software (for example, AlphaServer 2000).

The Tru64 UNIX Publications group cannot respond to system problems or technical support inquiries. Please address technical questions to your local system vendor or to the appropriate technical support office. Information provided with the software media explains how to send problem reports to Compaq.

The postpaid READER'S COMMENTS card requests the user's critical evaluation to assist in preparing future documentation.

Related Documents

The printed version of the Tru64 UNIX documentation set is coded with icons on the spines of books to help specific audiences quickly find the books that meet their needs.

Audience	Icon
General users	G
System and network administrators	S
Programmers	P
Device driver writers	D
Reference page users	R

Some books in the documentation set help meet the needs of several audiences. For example, the information in some system books is also used by programmers. Consider this when searching for information on specific topics. The *Documentation Overview,* Glossary, and Master Index provide information on all of the books in the Tru64 UNIX documentation set.

Associated DECevent Documentation

The following documents are associated with the DECevent documentation set.

Document	Order No
DECevent Event Management Utility for Tru64 UNIX Installation Guide	AA-QAA5C-TE
DECevent Analysis and Notification Utility for Tru64 UNIX User and Reference Guide	AA-QAA4C-TE
The DECevent Graphical User Interface User's Guide	AA-QE26C-TE

Chapter 1

The DECevent Event Management Utility Overview

This chapter discusses the DECevent event management utility, the translation of events, analysis and notification, and the DECevent graphical user interface.

1.1 Introduction

The DECevent event management utility provides the interface between a system user and the operating system's event logger. DECevent provides the following two main functions:

- Translation – DECevent allows a user to translate events into ASCII reports derived from system event entries (Bit-To-Text translations).
- Analysis and Notification – DECevent constantly monitors system events in an effort to isolate failing device components through analysis and can notify the proper individuals of a potential problem. Analysis and notification information can be found in the *DECevent Analysis and Notification Utility for Tru64 UNIX User and Reference Guide.*

1.2 DECevent and the Translation of Events

Users can request the translation of events from specific event logs into a variety of ASCII reports. The format and contents of the ASCII reports is determined by flags and parameters entered on the command line interface (CLI). The maximum length of the command line is limited to 255 characters.

DECevent translation has the following features:

- Translates event log entries into readable reports
- Specifies input and output sources
- Filters and selects input events

- Selects alternate report types
- Translates events as they occur

You also can maintain and customize the user environment with the interactive shell commands.

Necessary Privileges

Tru64 UNIX users need superuser privileges to use the translation and reporting features of DECevent, unless the event log file protection has been changed to allow all users to access the event log files.

1.3 DECevent Analysis and Notification

The added value function of DECevent provides constant monitoring of a system's event logger. When a significant number of events have occurred so that a DECevent threshold has been crossed, automatic analysis of the event is performed. Automatic analysis can result in the generation of Compaq-specific theory codes that enable Multivendor Customer Services to identify one or more failing field replaceable units (FRU).

Depending upon the theory codes generated from analysis, repair actions may be initiated. Also, depending upon the results of analysis, the proper individuals are notified of the event.

The analysis and notification flags allow the following features:

- Event analysis
- Event notification
- Customization of the DECevent environment

In addition to automatic analysis and notification, DECevent allows you to perform manual analysis on a user supplied event log. Manual analysis can also result in the generation of Compaq-specific theory codes that enable Multivendor Customer Services to determine a failing FRU.

Necessary Privileges

Tru64 UNIX users need superuser privileges to use the analysis and notification features of DECevent.

Note

A product authorization key (PAK) license is required to utilize the DECevent analysis and notification added-value options. The license is obtained by a Compaq Customer Services engineer through the technical information management architecture (TIMA) and must be installed on your system prior to kit installation. To find the DECevent service PAK in TIMA, select the TIMA

tools database and search for "Service PAKS." The title of the tool you need is "Service Tools Product Authorization Key (PAK) Distribution Tool." Refer to TIMA documentation for specifics.

In addition, customers may acquire the PAK information by requesting it from DSNlink hardware support. If DSNlink is installed on your system, type DSNLINK ITS at the system prompt and open the database articles for your operating system. Search for DECEVENT and select the article with the PAK information. You also can refer to the *DSNlink for OpenVMS Users's Guide* (AA-PBL4D-TE) for further information about DSNlink.

Licenses are verified at run time by the License Management Facility (LMF). For more information on LMF concepts and operation, enter the HELP LICENSE command at the DCL ($) prompt.

1.4 The DECevent Graphical User Interface

In addition to entering commands on the CLI, you have the option of using the DECevent graphical user interface (GUI) to perform translation and analysis. *The DECevent Graphical User Interface User's Guide* (AA-QE26A-TE) contains all the information necessary for you to use the GUI.

Chapter 2

The Help Command

This chapter discusses DECevent help and Tru64 UNIX help.

2.1 Overview

Help is provided for DECevent commands both through an internal DECevent help utility and through the Tru64 UNIX man utility.

2.2 DECevent Help

The DECevent *hlp* flag provides you with help based on a topic string. If the help utility contains information for that topic string, the information is presented. If no information is available for that topic string, a warning message is displayed and you are prompted to enter another topic string. Once help has been presented, you are prompted to enter another topic string. Each help topic can have zero or more subtopics that provide more information on the main topic.

2.2.1 Syntax

The following syntax is used for the DECevent *hlp* flag:

dia hlp [help_req]

Help_req Parameter

The *help_req* parameter is a topic string for which help has been requested.

Example

% dia hlp -b

This example provides help on the *-b* flag.

2.2.2 Exiting DECevent Help

There are two ways to exit from DECevent help. Either enter Ctrl/C followed by a carriage return, or type nothing at the topic prompt and enter a carriage return.

2.3 Tru64 UNIX Help

Help for the DECevent commands also is available through the Tru64 UNIX man utility. The following command shows how to obtain DECevent help using man:

```
% man dia
```

The man utility then displays a description of all DECevent commands.

Chapter 3

The DECevent Bit-To-Text Translation Feature

This chapter discusses the Bit-To-Text translation feature, including the necessary privileges, the command verb, translating event files, filtering input events, selecting alternative reports, and translating events as they occur.

3.1 Introduction

The DECevent utility enables you to produce Bit-To-Text ASCII reports derived from system event entries or user supplied event logs. The format of the ASCII report is determined by commands with flags, parameters, and selection keywords appended and entered on the command line interface (CLI). The maximum command line allowed is 255 characters. The DECevent bit-to-text feature performs the following:

- Translates event log files into readable reports
- Selects input and output sources
- Filters input events
- Selects alternate reports
- Translates events as they occur

Note

Refer to Appendix A for a list of all DECevent utility independent directory files.

3.2 Necessary Privileges

Tru64 UNIX users need superuser privileges to use the translation and reporting features of DECevent, unless the event log file protection privileges have been changed to allow all users to read the event log file.

3.3 Command Verb

The following DECevent utility command verb allows the translation of system event entries for Tru64 UNIX operating systems:

```
% dia -a
```

The *-a* flag is the default translation flag and does not need to be typed on the command line. Simply typing *dia* performs the same function as typing *dia -a* on the command line.

3.4 Translating Event Files

The DECevent utility uses the system event log file as the default input file. For Tru64 UNIX systems, the default file is /usr/adm/binary.errlog.

To produce a translated event report using the built-in defaults, use the following command:

```
% dia
```

This command by default produces a full report directed to the terminal screen, from the input event log file /usr/adm/binary.errlog.

The *-a* flag is understood on the command line and does not need to be entered. See Example 3–1 for an example of a full report.

3.4.1 Selecting an Alternate Input File

Use the following command to select an alternate input file for translation other than the default system event log file:

```
% dia -f errlog.sys
```

In the previous example, errlog.sys has been selected as the alternate file to be translated. You must precede the name of the input file with the *-f* flag.

The file must be a valid Tru64 UNIX file name.

3.4.2 Translating Multiple Input Files

DECevent can translate multiple input files, as shown in the following example:

```
% dia -f errlog.sys my_error.sys
```

3.4.3 Using Wildcard Characters

You can use asterisks (*) as wildcards to specify multiple input files. For example, if you want to translate all event log files located in the directory errlog, you could translate errlog_1.sys, errlog_2.sys, and errlog_3.sys with one file name, err*.sys, as shown in the following example:

```
% dia -f err*.sys
```

You also can use the wildcards to translate multiple event log files in multiple directories, as shown in the following example:

```
% dia -f errlog.sys my_error*.sys
```

3.4.4 Redirecting the Report to an Output File

To redirect the translated output to a file rather than to a terminal, enter the following command.

```
% dia > errlog_old.rpt
```

In the previous example, errlog_old.rpt is the output file into which the translated event information is written.

3.4.5 Reversing the Order of Input Events

To reverse the order of the input event log file being read by the DECevent utility, use the following command:

```
% dia -R
```

This command allows events contained in the default system event log file to be displayed in reverse chronological order, with the most recent events displayed first. The default is to display events in forward chronological order. Note that when the *-R* command is used the entry numbers in the report are listed 1-N.

Use the following command to display events contained in the event log file errorlog.sys in reverse chronological order:

```
% dia -R -f errorlog.sys
```

3.4.6 Creating a Binary Output File

Use the following command to create a smaller binary event log file from a larger event log file, using selection criteria. The following command creates a binary output file:

```
% dia -b error_sublog.bin
```

With this command the binary file error_sublog.bin is created from the default input system event log file. No text report output is generated.

3.5 Filtering Input Events

Sometimes you do not want all the information contained in the input event log file. The include (*-i*) and exclude (*-x*) flags allow you to filter input event log files to include or exclude event information.

3.5.1 Filtering Events by Event Types

To include only certain event types in the output report, use the *-i* flag, as shown in the following example:

```
% dia  -i disk=rz disk=ra92 cpu
```

In the previous example, only the RZ™ disks, RA92™ disks, and CPU entries are included in the output report. To exclude certain event types in the output report, use the *-x* flag, as shown in the following example:

```
% dia -x mem
```

In the previous example, memory entries are excluded from the output report.

Appendix D shows complete listings of all selection criteria for these flags.

3.5.2 Filtering Events by Date and Time

Date and time flags allow you to filter events by date and time occurrences. The date and time value is specified in the following format and defined in Table 3–1.

```
dd-mmm-yyyy[,hh:mm:ss]
```

Table 3–1 Date and Time Code Definitions

Time Code	Meaning	Example	Required
dd	Day of month	01	yes
mmm	Month	Jan	yes
yyyy	Year	1994	yes
hh	Hours	06	no
mm	Minutes	35	no
ss	Seconds	08	no

To select events between a certain time period on Tru64 UNIX systems, use the *-t* flag with the *s* and *e* parameters. You need both the *s* and *e* parameters on the command line to select events between certain time periods, as shown in the following example:

```
% dia -t s:15-jan-1993 e:20-jan-1993
```

To include events starting at a certain time for Tru64 UNIX systems, enter the following:

```
% dia -t s:15-jan-1993, 10:00
```

In the previous example, the output report will include all events that occurred after the date and time indicated. To include events before a certain time, enter the following command:

```
% dia -t e:15-jan-1993, 10:00
```

If no time is specified with a date, the default start time is midnight (00:00), and the default end time is 23:59:59.

3.5.3 Filtering Events by Entry Number

If the entry position within the event log file is known, a range of entries can be specified. In the following example, only entries 20 through 60 inclusive are translated.

```
% dia -e s:20 e:60
```

Either the *s* or *e* parameter can be omitted, but not both. If the *e* parameter is omitted, all entries from the starting entry indicated to the end of file are processed. If the *s* parameter is omitted, all events from the beginning through the end entry are processed.

3.6 Selecting Alternative Reports

The following sections describe different ways to produce reports.

3.6.1 Producing a Full Report

To produce a full report, use the *-o* flag with the full report type, as shown in the following example:

```
% dia -o full
```

The full report format provides a translation of all available information for each entry in the event log. The full report is the default report type and the flag does not need to be typed on the command line. Example 3–1 shows the format of a full report.

Example 3–1 Full Report Format

```
*************************** ENTRY  1  ****************************
Logging OS                                  2. Tru64 UNIX
System Architecture                         2. Alpha
Event sequence number                     838.
Timestamp of occurrence                       19-OCT-1993 23:30:47
Host name                                     alpha
System type register                x00000003 DEC 7000
Number of CPUs (mpnum)              x00000001
CPU logging event (mperr)           x00000000
Event validity                              1. O/S claims event is valid
Event severity                              3. High Priority
Entry type                                103. Tape Type Errors
---- Device Profile ----
Unit                                          30
Product Name                                  TA81 DSA Tape
---- MSCP Logged Msg ----
Logged Message Type Code                    2. Tape Message
Command Reference number            x00000000
Unit Number                                30.
MSCP Sequence number                        7.
Logged Message Format                       7. STI Drive Error
MSCP Flags                                 x41 Sequence Number Reset
                                               Operation Continuing
MSCP Unique Controller-ID   x000000000000FE01
MSCP Controller Model                       1. HSC50
MSCP Controller Class                       1. Mass Storage Controller
                                               class
Controller SW version                      40.
Controller HW version                       0.
MSCP Unique Unit-ID         x0000000000000C7B
MSCP Unit Model                             4. TA81
MSCP Unit Class                             3. Tape class
Unit SW version                             0.
Unit HW version                             0.
HSC Tape Event Code                     xFF6B Tape Drive Requested
                                               Error Log
Multiunit code                          x0022
```

Example 3–1 Full Report Format (Continued)

```
Gap count                                1.

Formatter SW version                    17.

Formatter HW version                     3.

TA81 SUB-SYSTEM
---------------

SPEED (IPS)                             25.

DENSITY                                 x04 GCR-6250

MSCP UNIT NUMBER                        30.

GAP COUNT                                0.

TRANSFER DESC BYTE 1                    x07 Write error

TRANSFER DESC BYTE 2                    x00

TRANSFER DESC BYTE 3                    x00

TRANSFER DESC BYTE 4                    x00

DRIVE EXT SENSE BYTE 1                  x0A Unit check
                                            Data check

DRIVE EXT SENSE BYTE 2                  x00

DRIVE EXT SENSE BYTE 3                  x01 Device interrupt check

DRIVE EXT SENSE BYTE 4                  x05 Tape moved
                                            Unrecoverable

DRIVE EXT SENSE BYTE 5                  x89 Formatter command code

DRIVE EXT SENSE BYTE 6                  xA4 Start/stop mode
                                            Auto speed mode
                                            GCR mode

DRIVE EXT SENSE BYTE 7                  x00

DRIVE EXT SENSE BYTE 8                  x00

DRIVE EXT SENSE BYTE 9                  x00

DRIVE EXT SENSE BYTE 10                 x00

DRIVE EXT SENSE BYTE 11                 x00

DRIVE EXT SENSE BYTE 12                 xC2 BOT

                                            Online
                                            Ready

DRIVE EXT SENSE BYTE 13                 x14 S/S mode
                                            GCR

DRIVE EXT SENSE BYTE 14                 x00

DRIVE EXT SENSE BYTE 15                 x00 Device command code

DRIVE EXT SENSE BYTE 16                 xFA Device marginal condition
                                            code

DRIVE EXT SENSE BYTE 17                 x00 Device flt/test
                                            completion code

DRIVE EXT SENSE BYTE 18                 x00 Device sub-flt/test
                                            completion code

DRIVE EXT SENSE BYTE 19                 x00
```

3.6.2 Producing a Brief Report

To produce a brief report, use the *-o* flag with the brief report type, as shown in the following example:

```
% dia -o brief
```

The brief report format provides translation of key information for each entry in the event log. Example shows the format for a brief report.

Example 3–2 Brief Report Format

```
************************ ENTRY    1 ********************************
Logging OS                            2. Tru64 UNIX
System Architecture                   2. Alpha
Event sequence number               838.
Timestamp of occurrence                  19-OCT-1993 23:30:47
Host name                                alpha
System type register          x00000003 DEC 7000
Number of CPUs (mpnum)        x00000001
CPU logging event (mperr)     x00000000
Event validity                        1. O/S claims event is valid
Event severity                        3. High Priority
---- Device Profile ----
Unit                                     30
Product Name                             TA81 DSA Tape
Logged Message Type Code              2. Tape Message
MSCP Flags                          x41  Sequence Number Reset
                                         Operation Continuing
HSC Tape Event Code               xFF6B Tape Drive Requested Error Log
```

3.6.3 Producing a Terse Report

To produce a terse report, use the *-o* flag with the terse report type, as shown in the following example:

```
% dia -o terse
```

The terse report format provides binary event information and displays register values and other ASCII messages in a condensed format. Example shows the format for a terse report.

Example 3–3 Terse Report Format

```
********************** ENTRY  1 **********************
Logging OS                                          2.
System Architecture                                 2.
Event sequence number                             838.
Timestamp of occurrence                 1993101923304700
Host name                                        alpha
System type register                          x00000003
Number of CPUs (mpnum)                        x00000001
CPU logging event (mperr)                     x00000000
Event validity                                      1.
Event severity                                      3.
Entry type                                        103.
---- Device Profile ----
Unit                                               30
Product Name                               TA81 DSA Tape
---- MSCP Logged Msg ----
Logged Message Type Code                            2.
Command Reference number                      x00000000
Unit Number                                        30.
MSCP Sequence number                                7.
Logged Message Format                               7.
MSCP Flags                                        x41
MSCP Unique Controller-ID                         x00
MSCP Controller Model                               1.
MSCP Controller Class                               1.
Controller SW version                              40.
Controller HW version                               0.
MSCP Unique Unit-ID                               x00
MSCP Unit Model                                     4.
MSCP Unit Class                                     3.
Unit SW version                                     0.
Unit HW version                                     0.
HSC Tape Event Code                             xFF6B
Multiunit code                                  x0022
Gap count                                           1.
```

Example 3–3 Terse Report Format (Continued)

```
Formatter SW version                                17.
Formatter HW version                                 3.

TA81 SUB-SYSTEM
-------------------

SPEED (IPS)                                         25.
DENSITY                                             x04
MSCP UNIT NUMBER                                    30.
GAP COUNT                                            0.
TRANSFER DESC BYTE 1                                x07
TRANSFER DESC BYTE 2                                x00
TRANSFER DESC BYTE 3                                x00
TRANSFER DESC BYTE 4                                x00
DRIVE EXT SENSE BYTE 1                              x0A
DRIVE EXT SENSE BYTE 2                              x00
DRIVE EXT SENSE BYTE 3                              x01
DRIVE EXT SENSE BYTE 4                              x05
DRIVE EXT SENSE BYTE 5                              x89
DRIVE EXT SENSE BYTE 6                              xA4
DRIVE EXT SENSE BYTE 7                              x00
DRIVE EXT SENSE BYTE 8                              x00
DRIVE EXT SENSE BYTE 9                              x00
DRIVE EXT SENSE BYTE 10                             x00
DRIVE EXT SENSE BYTE 11                             x00
DRIVE EXT SENSE BYTE 12                             xC2
DRIVE EXT SENSE BYTE 13                             x14
DRIVE EXT SENSE BYTE 14                             x00
DRIVE EXT SENSE BYTE 15                             x00
DRIVE EXT SENSE BYTE 16                             xFA
DRIVE EXT SENSE BYTE 17                             x00
DRIVE EXT SENSE BYTE 18                             x00
DRIVE EXT SENSE BYTE 19                             x00
```

3.6.4 Producing a Summary Report

To produce a summary report, use the *-o* flag with the summary report type, as shown in the following example:

```
% dia -o summary
```

The summary report format provides a statistical summary of the event entries in the event log.

Example 3–4 shows the format for a summary report.

Example 3–4 Summary Report Format

```
SUMMARY OF ALL ENTRIES LOGGED ON NODE alpha

unknown major class

MSCP                                    46.

SCSI                                     4.
```

3.7 Translating Events as They Occur

The *-c* flag allows events to be monitored as they occur in real time. This enables you to see the translated events immediately on the terminal, or to send translated events to an output file.

To monitor the event logger on Tru64 UNIX systems, enter the following command:

```
% dia -c
```

To send translated events to an output file instead of viewing the events on a terminal screen, enter the following command:

```
% dia -c -o brief > brief.rpt
```

The previous command creates a brief report called brief.rpt.

Using the *-o* brief report type with the *-c* flag is strongly recommended. Using the *-o* summary report type with the *-c* flag is not allowed.

3.8 Halting the Continuous Display of Events

To halt the continuous display of events as they occur, enter Ctrl/C. This stops the display, and the system prompt appears on the screen.

Chapter 4

Customizing Your Environment

This chapter discusses commands to customize your environments.

4.1 Commands to Customize Your Environments

You can customize your system environment from within the interactive command shell. The customized settings must be saved before exiting the interactive command shell. Examples of setting and saving customized settings follow.

4.1.1 Customizing the Default Event Log File

To set the event log to a file other than the default file, enter the following command:

```
dia> set evt /error/error_log.old
```

4.1.2 Customizing the Locale Parameter

To set the default locale in the DECevent utility, enter the following command:

```
dia> set loc AMERICAN_ENGLISH
```

Note

Only the AMERICAN_ENGLISH locale file is supported by DECevent utility.

4.2 Saving Customized Settings

To save your custom settings, enter the following command:

```
dia> sav
```

Note

You must save the customized settings before exiting the interactive command shell or the system default settings become valid again.

4.3 Restoring Customized Settings

To restore the customized settings in the DECevent utility, enter the following command:

```
dia> res
```

This uses the settings in your local settings file:

$HOME/FMG_ LOCAL_PARAM_LIBRARY.KNL

4.4 Restoring Default System Settings

To restore default system settings in the DECevent utility enter the following command:

```
dia> res sys
```

This uses the global settings in the following file:

$DIA_LIBRARY/FMG_GLOBAL_PARAM_ LIBRARY.KNL

Refer to Appendix B for a list of all default system settings.

Chapter 5

The DECevent dia Command Verb

This chapter discusses the DECevent *dia* command verb and the four main flags you can append to it.

5.1 Introduction

The *dia* DECevent command verb allows the translation of an event file residing on a Tru64 UNIX system. DECevent allows you to append four main flags to the *dia* command verb, each accomplishing different functions on an input event file. The four main flags are described in Table 5–1.

Table 5–1 DECevent Main Flags

Main	Flag Description
-a	The default qualifier for the dia command allowing the translation of events into a report.
-b	Allows smaller binary event log files to be created from larger event log files.
-c	Allows events to be formatted as they are logged by the operating system event logger.
-d	Allows the canonical format of events to be output in a hexidecimal dump format.

5.2 The dia -a Command

The *dia -a* command option performs a Bit-To-Text translation on the default system event file or on a user specified file if the *-f* flag is used. The default system event log file on a Tru64 UNIX operating system is /usr/adm/binary.errlog.

The *dia* command defaults to the *-a* flag if no main flag is specified on the command line. The *dia -a* command is the equivalent of the *dia* command. The following syntax is used for the *dia -a* command option:

```
dia [-a -f infile[ ...]]
```

5.2.1 Flags and Parameters for the -a Flag

The following flags can be appended to the *dia -a* command to further expand the utility function. Refer to Appendix C for a definition of these flags.

-f infile [...]
-v
-R
-e [s:start_number][e:end_number]
-i keyword [=val] [...]
-x keyword [=val] [...]
-H hostname [...]
-t [s:time][e:time]
-o output_type
> outfile

5.2.2 The infile Parameter

The *dia -a* command allows you to use the optional infile parameter. This allows you to choose one or more alternative input event files for translation. Reporting is done in sequential order. If you do not supply a file name for this parameter, the default event file is used. The default event file is defined as either the default system event log for each operating system, or a file specified using the *set evt* command.

The default system event log file on a Tru64 UNIX system is /usr/adm/binary.errlog.

5.2.3 Example

The following example results in the translation of events from the binary_errlogold.sys event file:

```
% dia -f /usr/adm/binary_errlogold.sys
```

5.3 The dia -b Command

The *dia --b* command allows you to copy all or part of a log file into another binary output file. This command is typically used in conjunction with the *-i* and *-x* flags and with selection keywords to select only those entries of interest. The binfile is the output file created from the *-b* command and is not optional.

The following syntax is used for the *dia -b* command option:

```
dia -b binfile [-f infile[ ...]]
```

5.3.1 Flags and Parameters for the -b Flag

The following list presents the valid flags and parameters for the *dia -b* command. Refer to Appendix C for definitions of these flags.

-f infile [...]
-v
-R
-j [rejfile]
-e [s:start_number][e:end_number]
-i keyword [=val] [...]
-x keyword [=val] [...]
-H hostname [...]
-t [s:time][e:time]

5.3.2 The binfile Parameter

The *dia -b* command creates a binary output file using the binfile parameter. You must supply a name for the binary output file with the .bin extension, as shown in the following example.

5.3.3 Example

The following example selects disk entries from the input file errlogold.sys and creates the disk.bin file:

```
% dia -b disk.bin -f errlogold.sys -i disk
```

5.4 The dia -c Command

The *dia -c* command reads and displays events as they occur directly from the system event logger. The output goes to the user terminal by default unless it is redirected to a file.

The following syntax is used for the *dia -c* command option:

```
dia -c
```

Note

A special file is created in the /tmp directory when you use the *dia -c* command: DECevent_MbxYYYY, where YYYY is a four digit number assigned by the system.

Do not delete this file while DECevent is running. These files are deleted upon normal termination of the *dia -c* command.

5.4.1 Flags and Parameters for the -c Flag

The following list presents the valid flags and parameters for the *dia -c* command. Refer to Appendix C for definitions of these flags.

-i keyword [=val] [...]
-x keyword [=val] [...]
-o output_type
> outfile

5.4.2 Examples

The following example reads events in real time and displays them on screen in the brief report format.

```
% dia -c -o brief
```

5.4.3 Halting Continuous Display Mode

To halt the continuous display of events as they occur, enter Ctrl/C. This stops the display, and the system prompt appears on the screen.

5.5 The dia -d Command

The *dia -d* command provides a brief report type followed by a dump of a generic buffer. The following syntax is used for the *dia -d* command option:

```
dia -d
```

5.5.1 Flags and Parameters for the -d Flag

The following list presents the valid flags and parameters for the *dia -d* command. Refer to Appendix C for definitions of these flags.

-f infile [...]
-v
-R
-e [s:start_number][e:end_number]
-i keyword [=val] [...]
-x keyword [=val] [...]
-H hostname [...]
-t [s:time][e:time]
> outfile

5.5.2 The infile Parameter

The *dia -d* command allows you to use the optional [infile] parameter. This allows you to choose one or more alternative input event files for translation. Reporting is done in sequential order. If you do not supply a file name for this parameter, the default event file is used. The default file is defined as either the default system event log for each operating system, or a file specified using the *set evt* command.

The default system event log file on a Tru64 UNIX system is /user/adm/binary.errlog.

5.5.3 Examples

The following example provides an ASCII output file called errlog.dmp containing disk entries from the errlogold.sys input file.

```
% dia -d -f errlogold.sys -i disk > errlog.dmp
```

Chapter 6

The ls Commands

This chapter discusses the DECevent *ls* commands.

6.1 Description

The DECevent *ls* commands allow you to display all requested rulesets listed in the specified knowledge library. Rulesets contain instructions necessary for the translation of events. A knowledge library contains the rulesets.

Table 6–1 lists each DECevent ls command.

Table 6–1 The ls Commands

Command	Action
ls evt	Lists all event rulesets.
ls can	Lists all canonical rulesets

6.2 The ls evt Command

The DECevent *ls evt* command lists all event rulesets contained in $DIA_LIBRARY/FMG_ETC__DEF_RULE_LIB.KNL.

Syntax

The following syntax is used for the *ls evt* command:

```
dia ls evt
```

This command provides a directory listing of the rulesets in the event knowledge library similar to the one shown in shown in Example 6–1.

Example 6–1 Event Knowledge Library List

```
Knowledge Library:

Ruleset Name        Path

=======================================

ARCHCTRL                        ROOT

HEADER_EV                       ROOT

                    .

                    .

                    .

SCSI2_DISP                      ROOT.ARCHCTRL

UNKNOWN_DEV_ERR_TIM_ATT         ROOT.ARCHCTRL

XMI_DISP                        ROOT.ARCHCTRL
```

6.3 The ls can Command

The DECevent *ls can* command allows you to list all canonical rulesets necessary for formatting a report.

The DECevent *ls can* command allows you to list all canonical rulesets contained in $DIA_LIBRARY/FMG_RPT__DEF_RULE_LIB.KNL.

Syntax

The following syntax is used for the ls can command:

```
dia ls can
```

This command provides a directory listing of the rulesets in the canonical knowledge library similar to the example shown in Example 6–2.

Example 6–2 Canonical Knowledge Library List

```
Knowledge Library:

Ruleset      Path
Name

=====================================================================

HEADER_CA    ROOT

MSCP_CA      ROOT

AXP_CA       ROOT

          .

          .

          .

KZMSA_CA     ROOT.HEADER_CA.HD_EVT_CA.IO_SUBSYS.IO_SUBSYS_DISP.IO_XMI

DEFAA        ROOT.HEADER_CA.HD_EVT_CA.IO_SUBSYS.IO_SUBSYS_DISP.ADAPTER

DEFEA        ROOT.HEADER_CA.HD_EVT_CA.IO_SUBSYS.IO_SUBSYS_DISP.ADAPTER

DEFTA        ROOT.HEADER_CA.HD_EVT_CA.IO_SUBSYS.IO_SUBSYS_DISP.ADAPTER

DEFZA        ROOT.HEADER_CA.HD_EVT_CA.IO_SUBSYS.IO_SUBSYS_DISP.ADAPTER
```

Chapter 7

The shw Commands

This chapter discusses the DECevent *shw* commands.

7.1 Description

The DECevent *shw* commands allow you to view a specific item, depending on the specific *shw* command issued. Table 7–1 lists each DECevent *shw* command.

Table 7–1 The shw Commands

Flag	Action
shw sel	Shows all values associated with selection information for the *-i* and *-x* flags.
shw sel [selection]	Shows only the single entry for the selection chosen.
shw set	Shows all possible settings.
shw set [setting]	Shows only the single entry for the setting chosen.

7.2 The shw sel Command

The DECevent *shw sel* command shows all keywords associated with selection information for the *-i* and *-x* flags.

Syntax

The following syntax is used for the shw sel command:

```
dia shw sel
```

This command gives a list of selection information similar to the one in Example 7–1.

Example 7–1 Selection Information List

```
Selection Information:
Key                          Item-Name                      Value
================================================================
bugchecks                    ca_EVT_swi_minor_sub_clas      1
cache                        ca_EVT_cpu_minor_class         4
configurations               ca_EVT_swi_minor_class         3
control_entries              ca_EVT_swi_minor_sub_clas      3
cpus                         ca_EVT_cpu_minor_class
dates                        ca_hd_gen_x__event_time
device_errors                ca_EVT_ios_minor_class
device_number                DEF_SEL__unit_number
disks                        DEF_SEL__disk
environmental_entries        ca_EVT_swi_minor_class         1
hosts                        ca_hd_gen_t__scs_name
informationals               ca_EVT_swi_minor_class         9
ios                          ca_EVT_ios_minor_class
io_subsystems                ca_EVT_ios_minor_class
mchks                        ca_EVT_cpu_minor_class         1
machine_checks               ca_EVT_cpu_minor_class         1
memory                       ca_EVT_mem_minor_class
nodes                        ca_hd_gen_t__scs_name
os                           ca_hd_gen_b__fm_os_id_code
operating_systems            ca_hd_gen_b__fm_os_id_code
pwr                          ca_EVT_swi_minor_class         1
power                        ca_EVT_swi_minor_class         1
sequence_numbers             ca_hd_gen_w__errseq
swi                          ca_EVT_swi_minor_class
software_informationals      ca_EVT_swi_minor_class
sync_communications          DEF_SEL__sync_comm
tapes                        DEF_SEL__tape
unknown_entries              DEF_SEL__unknown
osf_entry                    ca_hd_evt_w__entry
```

7.3 The shw sel [selection] Command

When the *shw sel* command is issued with the [selection] parameter, the command shows only the single entry for the [selection] chosen. The [selection] must be spelled out in full and it must be a valid selection type.

Syntax

The following syntax is used for the *shw sel* command with a [selection] parameter:

```
dia shw sel [selection]
```

The [selection] Parameter

The [selection] parameter must be one of the selection keys shown in Example 7–1. These keys are displayed when the *shw sel* command has been issued.

Example

The following is an example of the *shw sel* command with a specific [selection] parameter:

```
% dia shw sel operating_systems
```

Abbreviation of the [selection] parameter is not allowed.

This *shw sel* command gives the message shown in Example 7–2.

Example 7–2 A shw sel environmental_entries Command Example

```
Selection Information:
Key                          Item-Name                 Value
============================================================
environmental_entries        ca_EVT_swi_minor_class    1
```

7.4 The shw set Command

The DECevent *shw set* command shows all possible setting parameters.

Syntax

The following syntax is used for the *shw set* command:

```
dia shw set
```

A partial output from this command is shown in Example 7–3.

Example 7–3 Current Settings Information

```
Current Settings Information:

Name                                Value

=========================================================

CTR__RUL_LIB                        CTR__DEF_RUL_LIB

CTR__SUM_LIB                        CTR__DEF_SUM_LIB

DEF_CTR_RULE                        ROOT.HEADER_CA

DEF_ETC_RULE                        ROOT.HEADER_EV.OSF_HEADER_EV

ETC__RUL_LIB                        ETC__DEF_RUL_LIB

ETC__SEL_LIB                        ETC__DEF_SEL_LIB

RPT__RUL_LIB                        RPT__DEF_RUL_LIB

HELP_FILE                           FMG_HELP_FILE

LOCALE                              AMERICAN_ENGLISH

KNL__LIB_PREFIX                     FMG_

KNL__LIB_SUFFIX                     .KNL

DEFAULT_REPORT                      FULL_RE

ANA__FAC_DB                         ANA__FAC_STATE_DB

ANA__PRM_DB                         ANA__PRM_STATE_DB

ANA__RUL_LIB                        ANA__DEF_RUL_LIB

ANA__TMP_DB                         ANA__TMP_STATE_DB

NOT__RUL_LIB                        NOT__DEF_RUL_LIB

NOT__MAIL_LIB                       NOT__DEF_MAIL_LIB

NOT__EXTERNAL_LIB                   NOT__DEF_EXTERNAL_LIB

FMG__CUST_PROFILE                   /usr/sbin/DIA121/FMGPROFILE

FMG_VERSION                         2.1

SICL_STATE                          OFF

SICL_PROTOCOL_VERS                  1

AUTO_COPY                           OFF

ANALYSIS_TRIGGER_AUTOMATIC          AUTOMATIC

ANALYSIS_TRIGGER_MANUAL             MANUAL

NOT__DEF_CSC_PHONE                  1-800-354-9000

NOT__MAX_DSNLNK_MSG                 59

CMD_TRACE                           0
```

Example 7–3 Current Settings Information (Continued)

```
CTR_TRACE                       0
ETC_TRACE                       0
FMG__DEBUG_STATE                0
FMG__TRACE_STATE                0
RPT_TRACE                       0
TRACE_STATE                     0
CTR__DMP_RUL                    ROOT.DUMP_EVENT_CA
ETC__DMP_RUL                    ROOT.DUMP_EVENT_EV
CURRENT_CLI                     UNIX
VMS_DEFAULT                     /TRANSLATE
UNIX_DEFAULT                    -a
CHEROKEE_VMS_DEFAULT            /TRANSLATE
DEF_EVT_LOG                     /usr/adm/binary.errlog
PRINT_MESSAGES                  7
STREAM_SIZE                     20
SYS_ACT_LOG_NAME                ERT_ACTIVITY.LOG
KNL__DEF_ENTRIES                32
KNL__LIB_DIRECTORY              DIA_LIBRARY
VAX_VMS_STACK_SIZE              10
VAX_VMS_GUARD_SIZE              3
RULE_MAX_BYTECNT                100000
ANA_TRACE                       0
NOT_TRACE                       0
MAIL_DEBUG                      0
DEBUG_STATE                     1
ALPHA_VMS_STACK_SIZE            70
ALPHA_VMS_GUARD_SIZE            40
```

7.5 The shw set [setting] Command

When the *shw set* command is issued with a [setting] parameter, the command shows only the single entry for the setting chosen.

Syntax

The following syntax is used for the *shw set* command with a single [setting] parameter:

```
dia shw set [setting]
```

The [setting] Parameter

The [setting] parameter must be one of the valid setting names displayed when the *shw set* command was issued and must be entered in upper case.

Example

The following is an example of the *shw set* [setting] command:

```
% dia shw set DEFAULT_REPORT
```

This command displays the message shown in Example 7–4.

Example 7–4 A shw set default_report Command Example

```
Selection Information:
Name                    Value
================================================
DEFAULT_REPORT          FULL_RE
```

Chapter 8

The -int Flag

This chapter discusses the *-int* flag.

8.1 Description

The *-int* flag allows you to enter the DECevent interactive command shell. When you are in the DECevent interactive command shell, the **dia>** prompt is displayed on the screen. Chapter 4 provides a basic overview of the interactive command shell.

Note

All commands valid from the CLI also are valid from the interactive command shell without first entering the *dia* command verb. Some commands, such as the *set* commands, are valid only from within the interactive command shell and may not be entered from the CLI. If *-int* is specified on the command line, subsequent commands are ignored.

8.1.1 Entering the Interactive Command Shell

To enter the interactive command shell from a Tru64 UNIX system, at the system prompt, enter the following command:

```
% dia -int
```

The DECevent interactive command shell prompt, **dia>**, appears. The format for entering a command at the **dia>** prompt is the following:

```
dia> aaa xxxx yyyy
```

Where:

aaa is the command valid from the DECevent interactive command shell.
xxxx is the specific subject the command applies to.
yyyy is the parameter associated with the command.

Correct interactive command syntax is explained further in this chapter.

8.1.2 Exiting Interactive Mode

To exit from the interactive command shell, type `exit` at the **dia>** prompt followed by a carriage return.

8.1.3 Qualifiers and Parameters

Although there are no specific qualifiers or parameters for the *-int* flag, all commands valid from the CLI are valid from within the interactive command shell.

8.1.4 Interactive Command Examples

The following examples show how to enter the interactive command shell, how to issue a command from within the interactive command shell, and how to exit the interactive command shell.

Example: Entering the Interactive Command Shell

To enter the interactive command shell, enter the following command at the system prompt:

```
% dia -int
```

The DECevent interactive command shell prompt, **dia>**, then appears.

Example: Issuing a Command from the Interactive Command Shell

The following command shows how to issue a `set locale` command from **dia>**, the DECevent interactive prompt:

```
dia> set loc AMERICAN_ENGLISH
```

Note

All settings must be entered in uppercase letters.

The *set loc* command is described in detail in Section 8.2.2.

Example: Saving Settings from within the Interactive Command Shell

The results of all *set* commands must be saved before exiting the interactive command shell or the default settings once again become valid upon exiting the interactive command shell. To save settings, enter the following command at the **dia>** prompt:

```
dia> sav
```

Example: Exiting the Interactive Command Shell

To exit the interactive command shell, enter the following command at **dia>**, the interactive prompt:

```
dia> exit
```

8.2 Set, sav, and res Command Descriptions

The *set*, *sav*, and *res* commands work only from within the interactive command shell.

- The *set* commands allow you to customize or change system settings.
- The *sav* command allows you to save these changed settings.
- The *res* command allows you to restore previously set customer settings.

Table 8–1 lists each *set* command as well as the *sav* and *res* commands. Underlined parameters imply required input.

Table 8–1 The set, res, and sav Commands

Command	Action
set evt file	Allows you to select an event log file from which event reports are formatted.
set loc locale	Allows you to change the locale file for internationalization reasons.
res	Allows you to cancel customizations previously set in the customer local settings file.
sav	Allows you to store the current customization settings in a default file.

8.2.1 The set evt Command

The *set evt* command allows you to select an event log file from which event reports are formatted. This allows you, in the interactive command shell, to use an event log other than the default system event log without specifying the file name on all subsequent commands. Before exiting the interactive command shell, you must save this setting or system default settings become valid once again.

Syntax

The following syntax is used for the *set evt* command:

```
set evt errorlog.old
```

The file parameter

The file parameter is the name and path of the event log file used for formatting the report. You must enter a file parameter with this command.

Example

The following is an example of a *set evt* command:

```
dia> set evt /usr/users/binlog/binlog_oscar
```

Note

Set commands should be saved with the *sav* command. Refer to Section 8.2.4 for information of saving *set* command functions.

8.2.2 The set loc Command

The *set loc* command allows you to override the locale file previously set for the current system. This is done for internationalization reasons so natural language conventions can be added in the future.

Note

Only the AMERICAN_ENGLISH locale file will be supported for DECevent. However, documentation will be provided describing how to create new locale files.

Syntax

The following syntax is used for the *set loc* command:

```
set loc <locale>
```

locale Parameter

The locale parameter is a string defining the local language preferences available. An example is AMERICAN_ENGLISH. You must provide a locale parameter with this command.

Example

The following is an example of a set loc command:

```
dia> set loc AMERICAN_ENGLISH

dia> sav
```

Note

All settings must be entered in uppercase letters.

8.2.3 The res Command

The *res* command allows you to restore previous settings in the local customization file. It cancels customizations set since the last time the customization file was saved.

Syntax

The following syntax is used for the *res* command:

```
res
```

Example

The following is an example of a *res* command.

```
dia> res
```

8.2.4 The sav Command

The *sav* command allows you to store current customized settings in a default file. DECevent then uses these settings in subsequent sessions.

Syntax

The following syntax is used for the *sav* command:

```
sav
```

Example

The following is an example of the *sav* command:

```
dia> sav
```

Appendix A

DECevent Directories and Files

The following sections describe the necessary DECevent directories and files. The directories must contain the files listed for DECevent to perform correctly.

A.1 Image File

The main Tru64 UNIX image, *dia*, is located in the */usr/sbin/* directory.

A.2 man page Help

Tru64 UNIX man pages reside in the */usr/man/man8/dia.8* directory.

A.3 Interactive Help

DECevent interactive help physically resides in the */usr/opt/DIA201/lib/DIA* directory and has a symbolic link to */var/opt*.

A.4 Necessary DECevent Files

The environmental variable *DIA_LIBRARY* is defined to be */var/DIA*. The directory *var/DIA* has symbolic links to files that reside in the directory */var/opt/DIA230/DIA.* Files that reside in the directory */var/opt/DIA230/DIA* are linked to */usr/opt/DIA230/lib/DIA.* The files described in Table A–1 reside in the */usr/opt/DIA201/lib/DIA* directory. These files must be present in the directory for DECevent to perform correctly.

Table A–1 Necessary DECevent Files

File Name	File Description
FMG_AMERICAN_ENGLISH__HELP_STRINGS.KNL	Internal help library file in American English.
FMG_AMERICAN_ENGLISH__MESSAGES.KNL	Internal error message library file in American English.
FMG_HELP_FILE__AMERICAN_ENGLISH.HLP_OSF	Internal help information when *hlp* command is used in interactive mode.
FMG_FACTORY_GLOBAL_PARAM_ LIBRARY.KNL	Default settings library file provided for the main image.
FMG_GLOBAL_PARAM_LIBRARY.KNL	Present settings library file. Contains DECevent environmental settings.
FMG_DEF__SEL_RUL_LIB.KNL	Selection criteria library containing selection keys through which input events are filtered.
FMG_ETC__DEF_RUL_LIB.KNL	Operating system library file containing rules for converting the operating system specific events to the DECevent canonical form.
FMG_ETC__DEF_SEL_LIB.KNL	File containing rules for event to canonical translation.
FMG_CTR__DEF_RUL_LIB.KNL	File containing rules to translate the DECevent canonical events into readable text.
FMG_RPT__DEF_RUL_LIB.KNL	Report format library file containing rules for outputting different reports.
FMG_UNIX_COMMAND.KNL	Command style library file containing rules to parse UNIX commands.
FMG_ANA__DEF_RUL_LIB.KNL	File containing rules for analysis.
FMG_ANA__FAC_STATE_DB.KNL	File containing factory state analysis database.
FMG_ANA__PRM_STATE_DB.KNL	File containing permanent state analysis database.

Table A–1 Necessary DECevent Files (Continued)

File Name	File Description
FMG_NOT__DEF_EXTERNAL_LIB	File containing external notification list.
FMG_NOT__DEF_MAIL_LIB.KNL	File containing notification mailing lists.
FMG_NOT__DEF_RUL_LIB.KNL	File containing notification rules.
AXP_SYS12_EV5.BIN	Binary file containing analysis rules for AXP CPUs.
AXP_SYS17_EV4.BIN	Binary file containing analysis rules for AXP CPUs.
AXP_SYS27_EV45.BIN	Binary file containing analysis rules for AXP CPUs.
AXP_SYS9_EV4.BIN	Binary file containing analysis rules for AXP CPUs.
AXP_SYS9_EV5.BIN	Binary file containing analysis rules for AXP CPUs.
DEC_4000.BIN	Binary file containing analysis rules for the DEC 4000 CPU.
DEC_7000.BIN	Binary file containing analysis rules for the DEC 7000 CPU.
DSA.BIN	Binary file containing analysis rules for DSA devices.
HSAC.BIN	Binary file containing analysis rules for HSC devices.
KDM70.BIN	Binary file containing analysis rules for the KDM70 device.
RFXX.BIN	Binary file containing analysis rules for RF devices.

In addition to the files in the DECevent independent directory, a knowledge library file, FMG_LOCAL_PARAM_LIBRARY.KNL, is created in your home directory to be used when customized settings are saved. This library is created just by using DECevent. You need not be logged into your local directory for this library to be created.

Enter ls $HOME/*.KNL at the system prompt to see the library in your local directory.

Appendix B

System Settings

The following lists the system settings displayed with the *shw set* command and their default values for Tru64 UNIX.

```
Current Settings Information:
Name                              Value
============================================================
CMD_TRACE                         0
CTR_TRACE                         0
DEBUG_STATE                       0
ETC_TRACE                         0
FMG__DEBUG_STATE                  0
FMG__TRACE_STATE                  0
RPT_TRACE                         0
TRACE_STATE                       0
CTR__DMP_RUL                      ROOT.DUMP_EVENT_CA
CTR__RUL_LIB                      CTR__DEF_RUL_LIB
CTR__SUM_LIB                      CTR__DEF_SUM_LIB
DEF_CTR_RULE                      ROOT.HEADER_CA
DEF_ETC_RULE                      ROOT.HEADER_EV.OSF_HEADER_EV
ETC__DMP_RUL                      ROOT.DUMP_EVENT_EV
ETC__RUL_LIB                      ETC__DEF_RUL_LIB
ETC__SEL_LIB                      ETC__DEF_SEL_LIB
RPT__RUL_LIB                      RPT__DEF_RUL_LIB
HELP_FILE                         FMG_HELP_FILE
CURRENT_CLI                       UNIX
VMS_DEFAULT                       /TRANSLATE
UNIX_DEFAULT                      -a
CHEROKEE_VMS_DEFAULT              /TRANSLATE
DEF_EVT_LOG                       /usr/adm/binary.errlog
```

LOCALE	AMERICAN_ENGLISH
PRINT_MESSAGES	7
STREAM_SIZE	20
SYS_ACT_LOG_NAME	DIA_ACTIVITY.LOG
KNL__DEF_ENTRIES	32
KNL__LIB_DIRECTORY	DIA_LIBRARY
KNL__LIB_PREFIX	FMG_
KNL__LIB_SUFFIX	.KNL
VAX_VMS_STACK_SIZE	10
VAX_VMS_GUARD_SIZE	3
ALPHA_VMS_STACK_SIZE	50
ALPHA_VMS_GUARD_SIZE	30
RULE_MAX_BYTECNT	100000
DEFAULT_REPORT	FULL_RE
ANA_TRACE	0
NOT_TRACE	0
ANA__FAC_DB	ANA__FAC_STATE_DB
ANA__PRM_DB	ANA__PRM_STATE_DB
ANA__RUL_LIB	ANA__DEF_RUL_LIB
ANA__TMP_DB	ANA__TMP_STATE_DB
NOT__RUL_LIB	NOT__DEF_RUL_LIB
NOT__MAIL_LIB	NOT__DEF_MAIL_LIB
NOT__EXTERNAL_LIB	NOT__DEF_EXTERNAL_LIB
FMG__CUST_PROFILE	FMGPROFILE
FMG_VERSION	2.1
SICL_STATE	ON
SICL_PROTOCOL_VERS	1
AUTO_COPY	OFF
ANALYSIS_TRIGGER_AUTOMATI	AUTOMATIC
ANALYSIS_TRIGGER_MANUAL	MANUAL
NOT__DEF_CSC_PHONE	1-800-354-9000
NOT__MAX_DSNLNK_MSG	59
MAIL_DEBUG	0

User-Modifiable System Settings

Currently, the only user-modifiable system settings are the following:

- DEF_EVT_LOG
- LOCALE

Refer to Chapter 8 for instructions on how to change user modifiable settings and to Chapter 7 for information on how to show these settings.

Appendix C

Bit-To-Text Quick Reference

Table C–1 shows all the DECevent flags and qualifiers that are available for the Bit-To-Text translation of events.

Table C–1 DECevent Flags

Flag	Description
-a	Performs bit-to-text translation on event file.
-b binfile	Reads input file(s) and creates a single output file specified by binfile. Flag is mutually exclusive with -a, -c, and -d flags with -a being the default.
-c	Reads events directly from the error log deamon process. Flag is mutually exclusive with -a, -b, and -d flag with -a being the default.
-d	Produces output file in Hex format. Flag is mutually exclusive with -a, -b and -c flag with -a being the default.
-e [s:start_num] [e:end_num]	Selects event file entries by the position within the event file.
-F infile[...]	Provides for alternate input event log file. This parameter can be a single Tru64 UNIX file specification or a list of file specifications. Wildcards (*) are allowed.
-H hostname[...]	Selects event file entries by node name.
-i keyword[=val][...]	Includes event entries to be processed. The selection is by device class, entry type and/or device names.
-int	Places DECevent into interactive mode.
-j [rejfile]	Places all event entries failing selection criteria for a command into a specified file in binary format. Used only with the -b flag.

Table C–1 DECevent Flags (Continued)

Flag	Description
-o output_type	Output event in full, brief, terse, or summary report format. The default is full. Summary output type is not valid when used in conjunction with -c flag.
	full=All possible information.
	brief=Key information on each event entry.
	terse=Labeled information with no translations.
	summary=Statistical summary of event entries in the event file.
-R	Causes the event file to be read in reverse order.
-t [s:time][e:time]	Selects event file entries that have occurred in the time specified.
-v	Provides informational message about the number of entries selected and rejected while a file is being processed.
-x keyword=[val][...]	Excludes event entries from being processed. The selection is by device class, entry type, and or device names.
> outfile	Redirects the output from the default system output to the specified file (outfile).

All commands used at the command line interface also are valid within the interactive command shell. The following commands are valid only from within the interactive command shell:

- *set cmd*
- *set evt*
- *set loc*
- *res*
- *sav*
- *exit*

Appendix D

DECevent Selection Keywords for Bit-To-Text Translation

This appendix contains descriptions and examples of all selection keywords associated with Bit-To-Text translation of events.

D.1 The -i (include) Qualifier

The -i qualifier allows you to include event entries meeting the selection criteria specified. Only event entries meeting the selection criteria are included in the output.

Syntax

Syntax for the -i command is the following:

```
dia  -i keyword [= val] [ ...]
```

The val field is an optional field used to further define the selection keyword. For example, the keyword disk can be further defined with the value RZ23.

Note

All keyword values must be entered in upper case.

Example

```
%dia -i disk > filename.out
```

In the previous example, all entries selected from the event log are disk entries. The output is directed to the filename.out file.

Example

```
%dia -i disk=RZ23 > filename.out
```

In the previous example only RZ23 entries are selected from the event log. The output is

directed to a file named filename.out. You can combine -x and -i qualifiers in the same command line to further narrow the selection scope. An -i -i combination or an -x -x combination will result in an error.

Refer to Section D.4 for examples of using different include commands, and Section D.3 for the definitions of the selection keywords.

D.2 The -x (exclude) Qualifier

The -x qualifier allows you to exclude event entries meeting the criteria specified. Only event entries meeting the criteria are excluded from the output.

Syntax

Syntax for the -x command is the following:

```
dia  -x keyword [= val] [ ...]
```

The val field is an optional field used to further define the selection keyword. For example, the keyword disk can be further defined with the value RZ23.

Note

All keyword values must be entered in upper case.

Example

```
%dia -x disk > filename.out
```

In the previous example all entries in the log are selected except disk entries. The output is directed to the filename.out file.

Example

```
%dia -x disk=RZ23 > filename.out
```

In the previous example only RZ23 disk entries are excluded from the log. The output is directed to the filename.out file.

You can combine -x and -i qualifiers in the same command line to further narrow the selection scope. An -i -i combination or an -x -x combination will result in an error.

Refer to Section D.4 for examples of using different exclude commands, and Section D.3 for the definitions of the selection keywords.

D.3 The Selection Keywords and Their Definitions

You can use all keywords to exclude or include information from the output. The keywords and their definitions are listed in this section. The abbreviated forms of the keywords, bolded in the table, also are acceptable. For example, you may exclude environmental_entries with the following command:

```
%dia -x env
```

Table D–1 Keyword Definitions

Keyword	Event Type Definition
cache	Cache entries
cam	All SCSI entries logged by CAM logger.
configurations	Configuration entries
control_entries	System startup, or new errorlog creation
cpus	Machine check (670, 660, 630) entries for AXP
dates	Select on the timestamps in the entries (Use the -t qualifier instead)
device_errors	Device errors, device attention, device timeouts, logged message (MSCP), logged status (MSCP), logged MSCP messages
device_number	Entries that contain device numbers
disks	Disk class entries
environmental_entries	Power entries
hosts	Event logs with a node name (Use the -H qualifier instead)
kzmsa	Entries logged by CAM logger with CAM device class of XMI to SCSI.
kzpsa	Entries logged by CAM logger with CAM device class of SIMport adapters, PCI to SCSI.
kztsa	Entries logged by CAM logger with CAM device class of SIMport adapters, Turbochannel to SCSI.
swxcr	Entries logged by SWXCR.
informationals	Contain only logged message entries with the MSCP flags set for informational

Table D–1 Keyword Definitions (Continued)

Keyword	Event Type Definition
io_subsystems or **ios**	Device errors, device timeout, device attentions, logged status (MSCP), logged message (MSCP), logged MSCP message entries
machine_checks or **mch**ks	Events with machine checking information
memory	Events with soft error (CRD), extended (CRD), and memscan entries
nodes	Event logs with a host name (Use the -H qualifier instead)
operating_systems or **os**	Event logs with an operating system type
panic	Crash Re-start, System Panic, or User Panic entries
power or **pwr**	DEC 7000 CPU power entries
scsi_adapter	Entries logged by CAM logger with CAM device class of SCSI adapters, including local SCSI chip adapters, and SCSI bus adapters KZMSA, KZTSA, KZPSA, etc.
scsi_other	Entries logged by CAM logger other than disk, tape or processor.
scsi_processor	Entries logged by CAM logger with CAM device class of processor. Used only in the DECSAVE ASE environment.
sequence_numbers	Entries that contain an event sequence number
software_informationals or **swi**	Events with lastfail, system startup, system configuration, (volume mounts, volume dismounts, new errorlogs, timestamp entries)
sync_communications	Sync communication device entries
tapes	Event logs that contain all tape class entries
unknown_entries	Events with device types that have not been classified by the current set of rules
osf_entry	Events logged on a Tru64 UNIX operating system

D.4 Examples of Using the -i and -x Qualifiers

The following subsections provide -i and -x qualifier examples.

D.4.1 -i Qualifier Examples

The following example includes only power entries:

```
%dia -i power
```

The following example includes power, cpu, and tape entries:

```
%dia -i power cpu tape
```

The following example gives an error because two -i flags are not allowed:

```
%dia -i pwr -i cpu
```

D.4.2 -x Qualifier Examples

The following example excludes power entries and places the output in a file called outfile:

```
%dia -x pwr > outfile
```

The following example excludes power, cache and cpu entries and places the output in a file called outfile:

```
%dia -x pwr cpu cache> outfile
```

The following example gives an error because two -x flags are not allowed:

```
%dia -x pwr -x cpu
```

D.4.3 -x and -i Combinations

The following example includes all power entries that are not cpu entries:

```
%dia -i pwr -x cpu
```

The following example includes all power entries and excludes entries from node cxaiag that are not cpu entries:

```
%dia -i pwr -x cpu -H cxaiag
```

The following example includes all power and io_subsystem entries that are not cpu entries:

```
%dia -i pwr io_subsystem  -x cpu
```

The following example includes all power entries that are not cpu or software_informational entries:

```
%dia -i power -x cpu swi
```

The following example includes all power entries and excludes from the node cxaiag that are not cpu or software_informational entries:

```
%dia -i power -x cpu swi -H cxaiag
```