IEEE P1003.2a Draft 8 – December 1991

5.16 `man` — Display system documentation

5.16.1 Synopsis

`man [−k]` *name* ...

5.16.2 Description

The `man` utility shall write information about each of the *name* operands. If *name* is the name of a standard utility, `man` shall at a minimum write a message describing the syntax used by the standard utility, its options, and operands. If more information is available, the `man` utility shall provide it in an implementation-defined manner.

An implementation may provide information for values of *name* other than the standard utilities. Standard utilities that are listed as optional and that are not supported by the implementation either shall cause a brief message indicating that fact to be displayed or shall cause a full display of information as described previously.

5.16.3 Options

The `man` utility shall conform to the utility argument syntax guidelines described in 2.10.2.

The following option shall be supported by the implementation:

`−k` Interpret *name* operands as keywords to be used in searching a utilities summary database that contains a brief ''purpose'' entry for each standard utility and write lines from the summary database that match any of the keywords. The keyword search shall produce results that are the equivalent of the output of the following command:

```
grep -Ei '
name
name
...
' summary-database
```

(This assumes that the *summary-database* is a text file with a 7
single entry per line; this organization is not required and the 7
example using `grep −Ei` is merely illustrative of the type of 7
search intended.) The ''purpose'' entry to be included in the data- 7
base shall consist of a terse description of the purpose of the 7
utility. 7

5.16.4 Operands

The following operand shall be supported by the implementation:

name A keyword or the name of a standard utility. When `-k` is not 7
specified and *name* does not represent one of the standard utili- 7
ties, the results are unspecified.

5.16.5 External Influences

5.16.5.1 Standard Input

None.

5.16.5.2 Input Files

None.

5.16.5.3 Environment Variables

The following environment variables shall affect the execution of `man`:

LANG This variable shall determine the locale to use for the locale categories when both **LC_ALL** and the corresponding environment variable (beginning with **LC_**) do not specify a locale. See 2.6.

LC_ALL This variable shall determine the locale to be used to override any values for locale categories specified by the settings of **LANG** or any environment variables beginning with **LC_**.

LC_CTYPE This variable shall determine the locale for the interpretation of sequences of bytes of text data as characters (e.g., single- versus multibyte characters in arguments and in the summary database).

LC_MESSAGES This variable shall determine the language in which messages should be written.

PAGER This variable shall be interpreted as a string representing an output filtering command for writing the output to a terminal. Any string acceptable as a *command_string* operand to the `sh -c` command (see 4.56), shall be valid. When standard output is a terminal device, the manual-
page output shall be piped through the command. If the 7
PAGER variable is null or not set, the command shall be 7
either `more` (see 5.18) or another paginator utility docu- 8
mented in the system documentation. 8

5.16.5.4 Asynchronous Events

Default.

5.16.6 External Effects

5.16.6.1 Standard Output

The `man` utility writes text describing the syntax of the utility *name,* its options, and its operands, or, when `-k` is specified, lines from the summary database. The format of this text is implementation defined.

5.16.6.2 Standard Error

Used only for diagnostic messages.

5.16.6.3 Output Files

None.

5.16.7 Extended Description

None.

5.16.8 Exit Status

The `man` utility shall exit with one of the following values:

0 Successful completion.

>0 An error occurred.

5.16.9 Consequences of Errors

Default.

5.16.10 Rationale. *(This subclause is not a part of P1003.2a)*

Usage, Examples

The historical **MANPATH** variable is not included in POSIX because no attempt is made to specify naming conventions for manual page files, nor even to mandate that they are files at all. In some systems they could be a true database, or a hypertext file, or even fixed strings within the `man` executable. The developers of the standard consider the portability of manual pages to be outside their scope of work (and more appropriate to the working group developing applicationinstallation tools). However, users should be aware that **MANPATH** is implemented on a number of historical systems and that it can be used to tailor the search pattern for manual pages from the various categories (utilities, functions, file formats, etc.) when the system administrator reveals the location and conventions for manual pages on the system.

The keyword search can rely on at least the text of the clause titles from these utility descriptions and the implementation may add more keywords. The term ''clause titles'' refers to the strings such as

```
man - Display system documentation
ps - Report process status
```

History of Decisions Made

It is recognized that, as specified, the `man` utility is only of minimal usefulness. Opinion in the working group was strongly divided as to how much or how little information `man` should be required to provide. It was felt, however, that the provision of some portable way of accessing documentation would aid user portability. The arguments against a fuller specification were:

(1) Large quantities of documentation should not be required on a system cramped for disk space.

(2) The current manual system does not present information in a manner that greatly aids user-portability.

(3) A ''better help system'' is currently an area in which vendors feel that they can add value to their POSIX implementations.

The `−f` option was considered, but due to implementation differences, it was not included in the standard.

The description was changed to be more specific about what has to be displayed for a utility. The working group felt it was insufficient to allow a display of only the Synopsis, without giving a short description of what each option and operand does.

The ''purpose'' entry to be included in the database can be similar to the clause 7
title (less the numeric prefix) from this standard for each utility. These titles are 7
similar to those used in historical systems for this purpose. 7

See 4.40.10 for rationale concerning the default paginator. 7

5.17 `mesg` — Permit or deny messages

5.17.1 Synopsis

```
mesg [ y | n ]
```

5.17.2 Description

The `mesg` utility shall control whether other users are allowed to send messages
via `write`, `talk`, or other utilities to a terminal device. The terminal device 8
affected shall be determined by searching for the first terminal in the sequence of 8
devices associated with standard input, standard output, and standard error, 8
respectively. With no arguments, `mesg` shall report the current state without 8
changing it. Processes with appropriate privileges may be able to send messages
to the terminal independent of the current state.

5.17.3 Options

None.

5.17.4 Operands

The following operands shall be supported by the implementation in the POSIX Locale:

`y` Grant permission to other users to send messages to the terminal device. 8

`n` Deny permission to other users to send messages to the terminal device. 8

5.17.5 External Influences

5.17.5.1 Standard Input

None.

5.17.5.2 Input Files

None.

5.17.5.3 Environment Variables

The following environment variables shall affect the execution of `mesg`: