

NAME

stdbuf - change standard streams initial buffering

SYNOPSIS

stdbuf [-e *bufdef*] [-i *bufdef*] [-o *bufdef*] [*command* [...]]

DESCRIPTION

stdbuf is used to change the initial buffering of standard input, standard output and/or standard error streams for *command*. It relies on libstdbuf(3) which is loaded and configured by **stdbuf** through environment variables.

The options are as follows:

-e *bufdef*

Set initial buffering of the standard error stream for *command* as defined by *bufdef* (see *BUFFER DEFINITION*).

-i *bufdef*

Set initial buffering of the standard input stream for *command* as defined by *bufdef* (see *BUFFER DEFINITION*).

-o *bufdef*

Set initial buffering of the standard output stream for *command* as defined by *bufdef* (see *BUFFER DEFINITION*).

BUFFER DEFINITION

Buffer definition is the same as in libstdbuf(3):

"0" unbuffered

"L" line buffered

"B" fully buffered with the default buffer size

size fully buffered with a buffer of *size* bytes (suffixes 'k', 'M' and 'G' are accepted)

EXAMPLES

In the following example, the stdout stream of the `awk(1)` command will be fully buffered by default because it does not refer to a terminal. **stdbuf** is used to force it to be line-buffered so `vmstat(8)`'s output will not stall until the full buffer fills.

```
# vmstat 1 | stdbuf -o L awk '$2 > 1 || $3 > 1' | cat -n
```

SEE ALSO

libstdbuf(3), setvbuf(3)

HISTORY

The **stdbuf** utility first appeared in FreeBSD 8.4.

AUTHORS

The original idea of the **stdbuf** command comes from Pdraig Brady who implemented it in the GNU coreutils. Jeremie Le Hen implemented it on FreeBSD.