

NAME

stdbuf – Run COMMAND, with modified buffering operations for its standard streams.

SYNOPSIS

stdbuf *OPTION ... COMMAND*

DESCRIPTION

Run COMMAND, with modified buffering operations for its standard streams.

Mandatory arguments to long options are mandatory for short options too.

-i, --input=MODE

adjust standard input stream buffering

-o, --output=MODE

adjust standard output stream buffering

-e, --error=MODE

adjust standard error stream buffering

--help display this help and exit

--version

output version information and exit

If MODE is 'L' the corresponding stream will be line buffered. This option is invalid with standard input.

If MODE is 'O' the corresponding stream will be unbuffered.

Otherwise MODE is a number which may be followed by one of the following: KB 1000, K 1024, MB 1000*1000, M 1024*1024, and so on for G, T, P, E, Z, Y. In this case the corresponding stream will be fully buffered with the buffer size set to MODE bytes.

NOTE: If COMMAND adjusts the buffering of its standard streams ('tee' does for example) then that will override corresponding changes by 'stdbuf'. Also some filters (like 'dd' and 'cat' etc.) don't use streams for I/O, and are thus unaffected by 'stdbuf' settings.

EXAMPLES

tail -f access.log | stdbuf -oL cut -d ' ' -f1 | uniq

This will immediately display unique entries from access.log

BUGS

On GLIBC platforms, specifying a buffer size, i.e., using fully buffered mode will result in undefined operation.

AUTHOR

Written by Padraig Brady.

REPORTING BUGS

GNU coreutils online help: <<https://www.gnu.org/software/coreutils/>>

Report stdbuf translation bugs to <<https://translationproject.org/team/>>

COPYRIGHT

Copyright © 2018 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later <<https://gnu.org/licenses/gpl.html>>.

This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.

SEE ALSO

Full documentation at: <<https://www.gnu.org/software/coreutils/stdbuf>>
or available locally via: info '(coreutils) stdbuf invocation'