

# What does "2>&1" mean?

Asked 13 years, 10 months ago   Modified 6 months ago   Viewed 1.6m times



To combine `stderr` and `stdout` into the `stdout` stream, we append this to a command:

3037

`2>&1`



e.g. to see the first few errors from compiling `g++ main.cpp` :



`g++ main.cpp 2>&1 | head`

What does `2>&1` mean, in detail?

[bash](#)   [shell](#)   [unix](#)   [redirect](#)

Share Follow

edited Aug 10, 2022 at 19:30

asked May 3, 2009 at 22:57



[codeforester](#)

37.6k 16 107 132



[Tristan Havelick](#)

66.4k 20 54 64

62 @dbr I don't think it's just bash - I believe it's a bourne shell thing; hence sh, bash, ksh, ash, dash, etc. – [guns](#) May 3, 2009 at 23:49

8 This is part of the redirection paragraph describing POSIX-compliant shells, or POSIX shell for short. ksh is a POSIX shell for example. See: [pubs.opengroup.org/onlinepubs/009695399/utilities/...](https://pubs.opengroup.org/onlinepubs/009695399/utilities/...) – [jim mcnamara](#) Apr 4, 2013 at 2:55

19 This construct also works on Windows. – [Vadzim](#) Oct 22, 2013 at 13:45

7 It's generally better doing `2>&1` than `2>/dev/null ;-` – [F. Hauri - Give Up GitHub](#) Dec 8, 2013 at 12:11

15 I thought I'd mention that `|&` is shorthand for `2>&1 |` if you're using zsh. I can't speak to whether that applies to other bourne-like shells or if it's a zsh only feature. – [chrixian](#) Dec 17, 2013 at 5:20

19 Answers

Sorted by:

Highest score (default)



File descriptor 1 is the standard output ( `stdout` ).

File descriptor 2 is the standard error ( `stderr` ).

3383



At first, `2>1` may look like a good way to redirect `stderr` to `stdout` . However, it will actually be interpreted as "redirect `stderr` to a file named `1` ".