

IEEE Standards Interpretations for IEEE Std 1003.2™-1992 IEEE Standard for Information Technology--Portable Operating System Interfaces (POSIX®)-- Part 2: Shell and Utilities

Copyright © 1996 by the Institute of Electrical and Electronics Engineers, Inc. 3 Park Avenue New York, New York 10016-5997 USA All Rights Reserved.

Interpretations are issued to explain and clarify the intent of a standard and do not constitute an alteration to the original standard. In addition, interpretations are not intended to supply consulting information. Permission is hereby granted to download and print one copy of this document. Individuals seeking permission to reproduce and/or distribute this document in its entirety or portions of this document must contact the IEEE Standards Department for the appropriate license. Use of the information contained in this document is at your own risk.

IEEE Standards Department, Copyrights and Permissions, 445 Hoes Lane, Piscataway, New Jersey 08855-1331, USA

Interpretation Request #119

Topic: ed global commands **Relevant Clauses:** 4.20.7.3.7, 4.20.7.3.8, 4.20.7.3.25, 4.20.7.3.26

The indicated subclauses of the ed utility description do not indicate whether any given line in the buffer may be modified more than once during a single global command invocation. (A global command refers to any of the g, G, v, or V commands.) Lines 3758-3760 of subclause 4.20.7.3.7 describe the basic premise of the global commands: In the global command, the first step shall be to mark every line that matches the given BRE. Then, for every such line, the given command list shall be executed with the current line number set to the address of that line. Lines 3781-3783 of subclause 4.20.7.3.8 indicate that the commands list of the G command can modify any part of the buffer (this is also true for the other global commands): The commands read as part of the execution of the G command can address and affect any lines in the buffer. What comes into question is whether ed should allow a commands list to modify any one line in the buffer on more than one iteration of a global command.

Historically, the global commands do not execute the given command list on any marked line which has already been modified (via a substitute or move command in the command list), during the current global command invocation. For example, assume the following input file: ab bc ac ab And the following ed command: g/b/ 1,\$s/\$./ Every ed command I have found gives the following output: ab. bc. ac. ab. If the substitute command had been applied to the whole buffer (due to the 1,\$ address range) for each of the three /b/ matches found by the g command, the resulting file would be: ab... bc... ac... ab... This facet of the historical implementation prevents the extraordinary consequences possible if lines may be modified more than once from within any one global command. The following problematic examples assume an ed command that allows

these multiple modifications. (The input file is the same as in the example above.)

1) `g/b/ .,$s/b/x/g` This substitution would fail after the first iteration since the string to be substituted (`/b/`) has been removed.

2) `g/b/ $s/a/aaaaaaaa/g` This substitution would cause the last line to grow in length at an exponential rate. A similar case could split lines and make the file length grow exponentially.

3) `g/b/ +1m0` Here, the global search marked line 2 for later action, but on the first iteration, it was moved to the top of the file. Should the second iteration set dot to line 1 or line 4 before executing the command list? It is my opinion that the historical implementation of the global commands is safer and is at least more straight forward than the possible interpretation described above and should be clarified in the Relevant Clauses of POSIX.2.

Interpretation Response

The standard states the behavior for ed global commands and conforming implementations must conform to this. However, concerns have been raised concerning historical practice which are being referred to the sponsor.

Rationale for Interpretation

None.