

Ruby master - Feature #14715

Pathname#== should compare by #realpath instead of #to_s

04/26/2018 12:37 PM - hspazio (Fabio Pitino)

| | |
|---|--------|
| Status: | Open |
| Priority: | Normal |
| Assignee: | |
| Target version: | |
| Description | |
| Hi Core Team, | |
| I'd like to report a confusing behaviour I noticed with Pathname#== comparison method. | |
| See the following example: | |
| Supposed I've created a file "file.txt" on /home/user/ directory. | |
| <pre>require 'pathname' Pathname('file.txt').file? # => true, as expected Pathname('FILE.txt').file? # => true, it seems it's pointing to the same file above Pathname('FILE.txt').exist? # => true, seems correct Pathname('file.txt') == Pathname('FILE.txt') # => false # What's more interesting is: Pathname('file.txt').realpath == Pathname('FILE.txt').realpath # => false # Or: Pathname('file.txt') == Pathname('./file.txt') # => false</pre> | |
| Looking at the documentation (https://ruby-doc.org/stdlib-2.5.0/libdoc/pathname/rdoc/Pathname.html#method-i-3D-3D), The "=", "eq?" and "==" seem to have identical C implementation. What's the reason for that? Feels like the "==" makes sense to be strictly comparing String objects. However, as user of Pathname I'd expect the library to also make comparisons according to the domain rules of OS paths. | |
| Right now, to have the comparison between the files above returning "true" i have to do the following: | |
| <pre>Pathname('file.txt').realpath.to_s.downcase == Pathname('FILE.txt').realpath.to_s.downcase # => true</pre> | |
| This feels like playing with Pathname internals to do something that pathname should do out of the box. Perhaps a new method? | |
| At the OS level I if I do the following on my terminal: | |
| <pre>touch file.txt touch FILE.txt</pre> | |
| I'm editing the mtime of the same file. The OS knows I'm talking about the same file as it considers paths case insensitive. | |
| This does not seem to be the same for Pathname. | |

History

#1 - 08/26/2019 07:38 PM - jeremyevans0 (Jeremy Evans)

- Backport deleted (2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN)

- ruby -v deleted (ruby 2.4.0p0 (2016-12-24 revision 57164) [x86_64-darwin15])

- Tracker changed from Bug to Feature