Ruby master - Bug #14695

[2.5.1] `===` is 1.77x slower than `match?`

04/18/2018 07:30 AM - baweaver (Brandon Weaver)

Status:	Closed		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:	ruby 2.5.1p57 (2018-03-29 revision 63029) [x86_64-darwin17]	**Backport:**	2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN

Description

Was evaluating some of the === implementations while testing and came across this one:

```
# ruby 2.5.1p57 (2018-03-29 revision 63029) [x86_64-darwin17]

require 'benchmark/ips'

def run_benchmark(title, **benchmarks)
  puts '', title, '=' * title.size, ''

  # Validation
  benchmarks.each do |benchmark_name, benchmark_fn|
    puts "#{benchmark_name} result: #{benchmark_fn.call()}"
  end

  puts

  Benchmark.ips do |bm|
    benchmarks.each do |benchmark_name, benchmark_fn|
      bm.report(benchmark_name, &benchmark_fn)
    end

    bm.compare!
  end
end

regex = /foo/
# => /foo/
string = 'foobarbaz'
# => "foobarbaz"

run_benchmark('=== vs match? - 2.5.1',
  '===':    -> { regex === string },
  'match?': -> { regex.match? string }
)

=== vs match? - 2.5.1
=====================

=== result: true
match? result: true

Warming up --------------------------------------
                 ===   173.435k i/100ms
              match?   233.124k i/100ms
Calculating -------------------------------------
                 ===      3.174M (± 1.6%) i/s -    15.956M in   5.029027s
              match?      5.626M (± 2.5%) i/s -    28.208M in   5.016991s

Comparison:
              match?:  5626170.1 i/s
                 ===:  3173659.6 i/s - 1.77x  slower
```

It appears that === is running a bit slower than match?, though there may be some concerns around old code potentially relying on regex set globals after === too in case statements, so not sure there.

Related issues:	
Related to Ruby master - Feature #8110: Regex methods not changing global var...	**Closed**

History

#1 - 04/18/2018 07:57 AM - shyouhei (Shyouhei Urabe)

baweaver (Brandon Weaver) wrote:

> It appears that === is running a bit slower than match?

Accurate observation. Regexp#match? was introduced as a "faster alternative" to ordinary matches; see also https://bugs.ruby-lang.org/issues/8110

#2 - 04/18/2018 07:57 AM - shyouhei (Shyouhei Urabe)

- Related to Feature #8110: Regex methods not changing global variables added

#3 - 07/29/2019 10:20 PM - jeremyevans0 (Jeremy Evans)

- Status changed from Open to Closed