

Ruby - Bug #20104

Regexp#match returns nil but allocates T_MATCH objects

12/30/2023 04:39 PM - jeremyevans0 (Jeremy Evans)

Status:	Closed	
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:	ruby 3.4.0dev (2023-12-30T03:14:38Z master 8e32c01742) [x86_64-openbsd7.4]	Backport: 3.0: DONTNEED, 3.1: DONTNEED, 3.2: DONTNEED, 3.3: DONE

Description

Between Ruby 3.2 and 3.3, behavior changed so that Regexp#match will allocate a T_MATCH object even when there is no match. Example code:

```
h = {}
GC.start
GC.disable
ObjectSpace.count_objects(h)
matches = h[:T_MATCH] || 0
md = /\A[A-Z]+\Z/.match('1')
ObjectSpace.count_objects(h)
new_matches = h[:T_MATCH] || 0
puts "/\A[A-Z]+\Z/.match('1') => #{md.inspect} generates #{new_matches - matches} T_MATCH objects"
```

Result with Ruby 1.9-3.2:

```
 /\A[A-Z]+\Z/.match('1') => nil generates 0 T_MATCH objects
```

Results with Ruby 3.3.0 and current master branch:

```
 /\A[A-Z]+\Z/.match('1') => nil generates 1 T_MATCH objects
```

This results in a measurable performance decrease for both Sinatra and Roda web applications, as reported at: https://old.reddit.com/r/ruby/comments/18sxtv9/ruby_330_performance_ups_and_downs/

Thanks to GitHub users kiskoza and tagliala for producing a minimal example showing this issue: <https://github.com/caxlsx/caxlsx/issues/336>

Associated revisions

Revision e12d4c654e3cb7a4473014610bc3bae41aaf811e - 01/01/2024 09:28 PM - luke-gru (Luke Gruber)

Don't create T_MATCH object if /regexp/.match(string) doesn't match

Fixes [Bug #20104]

Revision e12d4c654e3cb7a4473014610bc3bae41aaf811e - 01/01/2024 09:28 PM - luke-gru (Luke Gruber)

Don't create T_MATCH object if /regexp/.match(string) doesn't match

Fixes [Bug #20104]

Revision e12d4c65 - 01/01/2024 09:28 PM - luke-gru (Luke Gruber)

Don't create T_MATCH object if /regexp/.match(string) doesn't match

Fixes [Bug #20104]

Revision f585171a6b1d6c20b3c162fd59dc874510ed2a49 - 01/30/2024 11:29 AM - naruse (Yui NARUSE)

merge revision(s) e12d4c654e3cb7a4473014610bc3bae41aaf811e: [Backport #20104]

Don't create T_MATCH object if /regexp/.match(string) doesn't match

```
Fixes [Bug #20104]
---
re.c | 9 +++++---
test/ruby/test_regexp.rb | 12 ++++++++
tool/lib/envutil.rb | 8 ++++++
3 files changed, 26 insertions(+), 3 deletions(-)
```

Revision f585171a6b1d6c20b3c162fd59dc874510ed2a49 - 01/30/2024 11:29 AM - naruse (Yui NARUSE)

merge revision(s) e12d4c654e3cb7a4473014610bc3bae41aaf811e: [Backport #20104]

```
Don't create T_MATCH object if /regexp/.match(string) doesn't match
```

```
Fixes [Bug #20104]
---
re.c | 9 +++++---
test/ruby/test_regexp.rb | 12 ++++++++
tool/lib/envutil.rb | 8 ++++++
3 files changed, 26 insertions(+), 3 deletions(-)
```

Revision f585171a - 01/30/2024 11:29 AM - naruse (Yui NARUSE)

merge revision(s) e12d4c654e3cb7a4473014610bc3bae41aaf811e: [Backport #20104]

```
Don't create T_MATCH object if /regexp/.match(string) doesn't match
```

```
Fixes [Bug #20104]
---
re.c | 9 +++++---
test/ruby/test_regexp.rb | 12 ++++++++
tool/lib/envutil.rb | 8 ++++++
3 files changed, 26 insertions(+), 3 deletions(-)
```

History

#1 - 12/30/2023 05:57 PM - luke-gru (Luke Gruber)

So I looked into this and it's this commit that introduces that behavior: 7193b404a1a56e50f8046d0382914907020c1559

Edit: create PR here <https://github.com/ruby/ruby/pull/9398>. I should probably add test for this, I'm just seeing if CI passes first.

#2 - 01/04/2024 10:45 PM - luke-gru (Luke Gruber)

So my PR has been merged but I believe a backport is desirable for 3.3. I'm not sure who to ping for a backport request. Thanks!

#3 - 01/04/2024 11:43 PM - jeremyevans0 (Jeremy Evans)

- Status changed from Open to Closed

This is already marked for backport to 3.3, it just needs to be closed.

#4 - 02/01/2024 12:07 AM - naruse (Yui NARUSE)

- Backport changed from 3.0: DONTNEED, 3.1: DONTNEED, 3.2: DONTNEED, 3.3: REQUIRED to 3.0: DONTNEED, 3.1: DONTNEED, 3.2: DONTNEED, 3.3: DONE

ruby_3_3 f585171a6b1d6c20b3c162fd59dc874510ed2a49 merged revision(s) e12d4c654e3cb7a4473014610bc3bae41aaf811e.