

Ruby - Bug #17545

Calling dup on a subclass of Proc returns a Proc and not the subclass

01/15/2021 09:26 PM - lamont (Lamont Granquist)

Status:	Closed	
Priority:	Normal	
Assignee:	ko1 (Koichi Sasada)	
Target version:		
ruby -v:	ruby 2.7.0p0 (2019-12-25 revision 647ee6f091) [x86_64-darwin18]	Backport: 2.5: UNKNOWN, 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: UNKNOWN
Description Trivial example: <pre>[1] pry(main)> class Test < Proc; end => nil [3] pry(main)> Test.new { puts "foo" }.dup => #<Proc:0x00007f91a78913c8 (pry):3></pre> Also fairly trivial workaround: <pre>[4] pry(main)> class Test < Proc [4] pry(main)* def dup [4] pry(main)* self.class.new(&super) [4] pry(main)* end [4] pry(main)* end => :dup [5] pry(main)> Test.new { puts "foo" }.dup => #<Test:0x00007f91b8e73e38 (pry):9></pre> Seems like a bug though, Proc#dup is probably calling the ruby-c equivalent of Proc.new rather than self.class.new and bypassing subclassing		

Associated revisions

Revision 7e21b77dc6bfefaf331a0dbf89782303b8cda05d - 01/13/2022 08:43 AM - ko1 (Koichi Sasada)

T#dup (T < Proc) should return T's object

T#dup (T < Proc) returns Proc object (not T) from Ruby 1.9.
[Bug #17545]

Revision 7e21b77dc6bfefaf331a0dbf89782303b8cda05d - 01/13/2022 08:43 AM - ko1 (Koichi Sasada)

T#dup (T < Proc) should return T's object

T#dup (T < Proc) returns Proc object (not T) from Ruby 1.9.
[Bug #17545]

Revision 7e21b77d - 01/13/2022 08:43 AM - ko1 (Koichi Sasada)

T#dup (T < Proc) should return T's object

T#dup (T < Proc) returns Proc object (not T) from Ruby 1.9.
[Bug #17545]

History

#1 - 01/29/2021 09:03 AM - ko1 (Koichi Sasada)

- Status changed from Open to Assigned
- Assignee set to ko1 (Koichi Sasada)

This behavior is from ruby 1.9.
Matz, is it a bug?

```
@all-ruby -e 'class Test < Proc; end; p Test.new { puts "foo" }.dup.class'
```

```
:ok: 1.8
Test
:ok: 1.9 -- 98e27016c9 (2021-01-29T07:22:12Z)
Proc
```

#2 - 02/18/2021 08:13 AM - ko1 (Koichi Sasada)

Fix is easy:

<https://github.com/ruby/ruby/pull/4197>

#3 - 02/19/2021 02:00 AM - matz (Yukihiro Matsumoto)

It should be an instance of the subclass for consistency. But considering that we discourage the use of subclass for the standard class, the fix should not be applied soon. Maybe 3.2 or later.

Matz.

#4 - 01/13/2022 08:44 AM - ko1 (Koichi Sasada)

- *Status changed from Assigned to Closed*

Applied in changeset [git|7e21b77dc6bfefaf331a0dbf89782303b8cda05d](https://github.com/ruby/ruby/commit/7e21b77dc6bfefaf331a0dbf89782303b8cda05d).

T#dup (T < Proc) should return T's object

T#dup (T < Proc) returns Proc object (not T) from Ruby 1.9.

[Bug [#17545](#)]