

## Ruby master - Bug #9526

### Method#owner returns wrong owner if using prepend.

02/17/2014 02:24 PM - yui-knk (Kaneko Yuichiro)

<b>Status:</b> Rejected	
<b>Priority:</b> Normal	
<b>Assignee:</b>	
<b>Target version:</b>	
<b>ruby -v:</b> ruby-2.0.0-p353 [ x86_64 ] & ruby-2.1.0 [ x86_64 ]	<b>Backport:</b> 1.9.3: UNKNOWN, 2.0.0: UNKNOWN, 2.1: UNKNOWN
<b>Description</b>	
rdoc indicates that Method#owner will return a class or module that defines a method. <a href="http://www.ruby-doc.org/core-2.0/Method.html#method-i-owner">http://www.ruby-doc.org/core-2.0/Method.html#method-i-owner</a>	
<pre>module MyMod   def x(y)     super(y*y)   end end  class Blah   def x(y)     y + 1   end   prepend MyMod end  Blah.new.method(:x).owner # =&gt; MyMod</pre>	
In this case, a class or module that defines a method is Blah. So I think Blah.new.method(:x).owner should return Blah.	

### History

#### #1 - 02/18/2014 01:30 AM - shugo (Shugo Maeda)

Kaneko Yuichiro wrote:

In this case, a class or module that defines a method is Blah.  
So I think Blah.new.method(:x).owner should return Blah.

I don't catch your point.

Blah.new.x(1) calls MyMod#x, so Blah.new.method(:x).owner should return MyMod, shouldn't it?

#### #2 - 02/18/2014 01:30 AM - shugo (Shugo Maeda)

- Status changed from Open to Feedback

#### #3 - 02/18/2014 05:16 AM - Anonymous

Also for me, this is the expected behavior. MyMod is prepended before Blah, and therefore Blah.new.method( :x ) is taken from MyMod. As soon as you perform module MyMod; remove\_method( :x ) end, method x will be again taken from Blah.

#### #4 - 02/18/2014 01:46 PM - yui-knk (Kaneko Yuichiro)

Sorry, I misunderstood behavior of prepend. Shugo and Boris are right. Thanks for your quick response. Please close this ticket :)

#### #5 - 02/19/2014 09:09 AM - shugo (Shugo Maeda)

- Status changed from Feedback to Rejected

Kaneko Yuichiro wrote:

Sorry, I misunderstood behavior of prepend. Shugo and Boris are right. Thanks for your quick response. Please close this ticket :)

Thanks for your confirmation.