Ruby - Bug #7613

An Alias for a class method inherited from the Class class is not equal to the original method

12/24/2012 06:38 AM - zhangsu (Su Zhang)

Status:	Closed		
Priority:	Normal		
Assignee:			
Target version:	1.9.3		
ruby -v:	ruby 1.9.3p327 (2012-11-10) [i386-mingw32]	**Backport:**	

Description

```
class Stream
  class << self
    alias_method :open, :new
  end
end

open = Stream.method(:open)
new = Stream.method(:new)
p open, new                   # => #<Method: Stream.new>, #<Method: Class#new>
p open.receiver, new.receiver # => Stream, Stream
p open == new                 # => false
```

Expect the last line to return true, but got false. According to the documentation for Method#==:

> Two method objects are equal if they are bound to the same object and refer to the same method definition.

Here, open and new are all bound to Stream and I expect them to refer to the method definition too.

Associated revisions

Revision d5a39c0aa85a705fcf360a535a278aed9116f644 - 12/28/2012 02:23 AM - shugo (Shugo Maeda)

- proc.c (method_eq): fix the documentation to refer to owner.
  [ruby-core:51105] [Bug #7613]

- test/ruby/test_method.rb (test_alias_onwer): new test to confirm
  that `a == b' returns false if owners of a and b are different.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@38638 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision d5a39c0a - 12/28/2012 02:23 AM - shugo (Shugo Maeda)

- proc.c (method_eq): fix the documentation to refer to owner.
  [ruby-core:51105] [Bug #7613]

- test/ruby/test_method.rb (test_alias_onwer): new test to confirm
  that `a == b' returns false if owners of a and b are different.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@38638 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 12/28/2012 11:23 AM - shugo (Shugo Maeda)

- Status changed from Open to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r38638.
Su, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

- proc.c (method_eq): fix the documentation to refer to owner.
  [ruby-core:51105] [Bug #7613]

- test/ruby/test_method.rb (test_alias_onwer): new test to confirm
  that `a == b' returns false if owners of a and b are different.

#2 - 12/28/2012 11:26 AM - shugo (Shugo Maeda)

zhangsu (Su Zhang) wrote:

> ```
> class Stream
>   class << self
>     alias_method :open, :new
>   end
> end
>
> open = Stream.method(:open)
> new = Stream.method(:new)
> p open, new                   # => #<Method: Stream.new>, #<Method: Class#new>
> p open.receiver, new.receiver # => Stream, Stream
> p open == new                 # => false
> ```
>
> Expect the last line to return true, but got false. According to the documentation for Method#==:

open == new returns false because their owners are different.

```
p [open.owner, new.owner] #=> [#<Class:Stream>, Class]
```

> > Two method objects are equal if they are bound to the same object and refer to the same method definition.
>
> Here, open and new are all bound to Stream and I expect them to refer to the method definition too.

I've fixed the documentation of Method#== as follows:

> Two method objects are equal if they are bound to the same
> object and refer to the same method definition and their owners are the
> same class or module.

#3 - 12/28/2012 11:36 AM - shugo (Shugo Maeda)

shugo (Shugo Maeda) wrote:

> open == new returns false because their owners are different.
>
> ```
> p [open.owner, new.owner] #=> [#<Class:Stream>, Class]
> ```

I forgot to explain why Method#== returns false if owners are different.
If owners of methods are different, the behavior of super is different in the methods,
so Method#== should not return true for the methods.