

Ruby master - Bug #7842

An alias of a "prepend"ed method skips the original method when calling super

02/13/2013 10:53 PM - mame (Yusuke Endoh)

Status:	Closed	
Priority:	Normal	
Assignee:	nobu (Nobuyoshi Nakada)	
Target version:	2.1.0	
ruby -v:	ruby 2.0.0dev (2013-02-13 trunk 39225) [x86_64-linux]	Backport: 1.9.3: DONTNEED, 2.0.0: DONE

Description

Hello,

```
module P
  def m; puts "P"; super; end
end
class A
  def m; puts "A"; end
end
class B < A
  def m; puts "B"; end
  prepend P
  alias m2 m
end
B.new.m2
#=> expected: P, B, A
#=> actual:   P, A
```

Is this intentional?

It looks weird to me that calling super of P#m (as m2) skips A#m.

Yusuke Endoh mame@tsg.ne.jp

Related issues:

Related to Ruby master - Feature #7836: Need a way to get Method and UnboundM...	Closed	
Related to Ruby master - Bug #11189: alias prepended module	Closed	
Has duplicate Ruby master - Bug #9236: include + include + alias_method + sup...	Closed	12/10/2013
Has duplicate Ruby master - Bug #7993: owner of methods defined after using M...	Closed	03/01/2013

Associated revisions

Revision b71956d1 - 12/13/2013 01:29 PM - nobu (Nobuyoshi Nakada)

proc.c, vm_method.c: fix super and alias

- proc.c (method_owner): return the class where alias is defined, not the class original method is defined.
- vm_method.c (rb_method_entry_make, rb_alias): store the originally defined class in me. [Bug #7993] [Bug #7842] [Bug #9236]
- vm_method.c (rb_method_entry_get_without_cache): cache included module but not iclass.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@44175 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 44175 - 12/13/2013 01:29 PM - nobu (Nobuyoshi Nakada)

proc.c, vm_method.c: fix super and alias

- proc.c (method_owner): return the class where alias is defined, not the class original method is defined.
- vm_method.c (rb_method_entry_make, rb_alias): store the originally defined class in me. [Bug #7993] [Bug #7842] [Bug #9236]
- vm_method.c (rb_method_entry_get_without_cache): cache included module but not iclass.

Revision 44175 - 12/13/2013 01:29 PM - nobu (Nobuyoshi Nakada)

proc.c, vm_method.c: fix super and alias

- proc.c (method_owner): return the class where alias is defined, not the class original method is defined.

- `vm_method.c (rb_method_entry_make, rb_alias)`: store the originally defined class in me. [Bug #7993] [Bug #7842] [Bug #9236]
- `vm_method.c (rb_method_entry_get_without_cache)`: cache included module but not `iclass`.

Revision 44175 - 12/13/2013 01:29 PM - nobu (Nobuyoshi Nakada)

`proc.c, vm_method.c`: fix super and alias

- `proc.c (method_owner)`: return the class where alias is defined, not the class original method is defined.
- `vm_method.c (rb_method_entry_make, rb_alias)`: store the originally defined class in me. [Bug #7993] [Bug #7842] [Bug #9236]
- `vm_method.c (rb_method_entry_get_without_cache)`: cache included module but not `iclass`.

Revision 44175 - 12/13/2013 01:29 PM - nobu (Nobuyoshi Nakada)

`proc.c, vm_method.c`: fix super and alias

- `proc.c (method_owner)`: return the class where alias is defined, not the class original method is defined.
- `vm_method.c (rb_method_entry_make, rb_alias)`: store the originally defined class in me. [Bug #7993] [Bug #7842] [Bug #9236]
- `vm_method.c (rb_method_entry_get_without_cache)`: cache included module but not `iclass`.

Revision 44175 - 12/13/2013 01:29 PM - nobu (Nobuyoshi Nakada)

`proc.c, vm_method.c`: fix super and alias

- `proc.c (method_owner)`: return the class where alias is defined, not the class original method is defined.
- `vm_method.c (rb_method_entry_make, rb_alias)`: store the originally defined class in me. [Bug #7993] [Bug #7842] [Bug #9236]
- `vm_method.c (rb_method_entry_get_without_cache)`: cache included module but not `iclass`.

Revision 44175 - 12/13/2013 01:29 PM - nobu (Nobuyoshi Nakada)

`proc.c, vm_method.c`: fix super and alias

- `proc.c (method_owner)`: return the class where alias is defined, not the class original method is defined.
- `vm_method.c (rb_method_entry_make, rb_alias)`: store the originally defined class in me. [Bug #7993] [Bug #7842] [Bug #9236]
- `vm_method.c (rb_method_entry_get_without_cache)`: cache included module but not `iclass`.

Revision f0f3821b - 12/22/2013 02:33 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) 44175,44179: [Backport #9236]

```
* proc.c (method_owner): return the class where alias is defined, not
the class original method is defined.
```

```
* vm_method.c (rb_method_entry_make, rb_alias): store the originally
defined class in me. [Bug #7993] [Bug #7842] [Bug #9236]
```

```
* vm_method.c (rb_method_entry_get_without_cache): cache included
module but not iclass.
```

```
* proc.c (mnew_from_me): achieve the original defined_class from
prepeded iclass, to fix inherited owner.
```

```
* proc.c (method_owner): return the defined class, but not the
class which the method object is created from.
```

git-svn-id: [svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_0_0@44345](https://ci.ruby-lang.org/ruby/branches/ruby_2_0_0@44345) b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 02/14/2013 07:28 AM - marcandre (Marc-Andre Lafortune)

I agree, this can only be a bug.

#2 - 02/14/2013 04:14 PM - nobu (Nobuyoshi Nakada)

Rather I expect [B, A] in this case.

#3 - 02/14/2013 11:59 PM - mame (Yusuke Endoh)

Rather I expect [B, A] in this case.

Fine. I just wanted to point out the weird skipping.
(Actually, I expect nothing for such a code :-)

2013/2/14, SASADA Koichi ko1@atdot.net:

(2013/02/14 16:14), nobu (Nobuyoshi Nakada) wrote:

Rather I expect [B, A] in this case.

+1

--

// SASADA Koichi at atdot dot net

--

Yusuke Endoh mame@tsg.ne.jp

#4 - 02/15/2013 01:01 AM - marcandre (Marc-Andre Lafortune)

nobu (Nobuyoshi Nakada) wrote:

Rather I expect [B, A] in this case.

You do?

There would be no way to safely monkey patch this method without using Prepend! So any 1.9.x library that uses monkey patching like `alias_method_chain` could be incompatible with prepended classes!

I expect:

```
B.instance_method(:m) == B.instance_method(:m2) # => true in rc2, false in trunk, should be true
B.new.m2 # => [P, B, A], since they are aliases appart from their name
```

Here is another bug (even more clear) that might be related:

```
B.instance_method(:m).owner # => P in rc2 (ok), B in trunk (not ok)
B.instance_method(:m2).owner # => #, should be P
```

I wish I had time to look at it further.

#5 - 02/17/2013 11:54 PM - matz (Yukihiro Matsumoto)

I expect:

```
module P
  def m; puts "P"; super; end
end
class A
  def m; puts "A"; end
end
class B < A
  def m; puts "B"; end
  alias m2 m
  prepend P
  alias m3 m
end
B.new.m2
#=> expected: B, A
B.new.m3
#=> expected: P, B, A
```

Matz.

#6 - 02/18/2013 06:23 AM - phluid61 (Matthew Kerwin)

I'm sorry, but this ticket confuses me. Why does "A" ever get output, if B#m does not also call super?

#7 - 02/18/2013 12:21 PM - marcandre (Marc-Andre Lafortune)

phluid61 (Matthew Kerwin) wrote:

I'm sorry, but this ticket confuses me. Why does "A" ever get output, if B#m does not also call super?

You're right to be confused, there's a missing super in the examples that everyone assumed was there.

So here's the corrected version:

```
module P
  def m; puts "P"; super; end
end
class A
  def m; puts "A"; end
end
class B < A
  def m; puts "B"; super; end
  alias m2 m
  prepend P
  alias m3 m
end

B.new.m2 #=> expected: B, A
B.new.m3 #=> expected: P, B, A
```

#8 - 02/18/2013 12:22 PM - marcandre (Marc-Andre Lafortune)

- Category set to core
- Assignee deleted (matz (Yukihiro Matsumoto))

#9 - 02/22/2013 08:57 AM - ko1 (Koichi Sasada)

- Assignee set to nobu (Nobuyoshi Nakada)
- Target version changed from 2.6 to 2.1.0

marcandre: why you remove assignee?
you mean specification discussion was finished at [ruby-core:52386]#5?
I assume the reason and I assign this ticket nobu, the patch monster.

#10 - 02/22/2013 10:46 AM - marcandre (Marc-Andre Lafortune)

ko1 (Koichi Sasada) wrote:

marcandre: why you remove assignee?
you mean specification discussion was finished at [ruby-core:52386]#5?
I assume the reason and I assign this ticket nobu, the patch monster.

Exactly.

#11 - 12/13/2013 10:29 PM - nobu (Nobuyoshi Nakada)

- Status changed from Assigned to Closed
- % Done changed from 0 to 100

This issue was solved with changeset r44175.
Yusuke, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

proc.c, vm_method.c: fix super and alias

- proc.c (method_owner): return the class where alias is defined, not the class original method is defined.
- vm_method.c (rb_method_entry_make, rb_alias): store the originally defined class in me. [Bug #7993] [Bug #7842] [Bug #9236]
- vm_method.c (rb_method_entry_get_without_cache): cache included module but not iclass.

#12 - 12/18/2013 12:49 PM - nobu (Nobuyoshi Nakada)

- Backport set to 1.9.3: DONTNEED, 2.0.0: REQUIRED

#13 - 01/12/2014 06:57 AM - nagachika (Tomoyuki Chikanaga)

- Backport changed from 1.9.3: DONTNEED, 2.0.0: REQUIRED to 1.9.3: DONTNEED, 2.0.0: DONE

r44175 was already backported to ruby_2_0_0 at r44345.

#14 - 04/25/2014 06:17 AM - nobu (Nobuyoshi Nakada)

- Description updated

#15 - 05/28/2015 12:33 AM - usa (Usaku NAKAMURA)

- Related to Bug #11189: alias prepended module added