

Ruby - Bug #11425

super_method fails to find the super method when combined with Module#prepend

08/09/2015 08:10 PM - matsuda (Akira Matsuda)

Status:	Closed	
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:	ruby 2.3.0dev (2015-08-09 trunk 51513) [x86_64-darwin14]	Backport: 2.0.0: UNKNOWN, 2.1: UNKNOWN, 2.2: UNKNOWN

Description

super_method now fails to find the target method of Module#prepend maybe since [51501](#)
<https://github.com/ruby/ruby/commit/6b7c4df>

```
class C; def a() end end
module M def a() end end
C.prepend M
p C.instance_method(:a).super_method

ruby 2.2.2p95 (2015-04-13 revision 50295) [x86_64-darwin14]
#=> #<UnboundMethod: Object(C) #a>

ruby 2.3.0dev (2015-08-09 trunk 51513) [x86_64-darwin14]
#=> nil
```

History

#1 - 08/09/2015 08:22 PM - matsuda (Akira Matsuda)

- File bug11425.patch added

This happens only for UnboundMethod.

```
C.new.method(:a).super_method
```

correctly returns a Method.

Attached a patch with a failing test case.

#2 - 08/10/2015 02:42 PM - nobu (Nobuyoshi Nakada)

- Description updated

Another issue, UnboundMethod#inspect in 2.2 also seems wrong.
It should be "#<UnboundMethod: C#a>".

#3 - 08/10/2015 02:43 PM - nobu (Nobuyoshi Nakada)

- Description updated

#4 - 07/05/2019 12:25 AM - jeremyevans0 (Jeremy Evans)

- Status changed from Open to Closed

This was fixed between Ruby 2.3 and 2.4:

```
$ ruby23 -e '
> class C; def a() end end
> module M def a() end end
> C.prepend M
> p C.instance_method(:a).super_method
> '
nil
$ ruby24 -e '
> class C; def a() end end
```

```
> module M def a() end end
> C.prepend M
> p C.instance_method(:a).super_method
> '
#<UnboundMethod: C#a>
```

Files

bug11425.patch	935 Bytes	08/09/2015	matsuda (Akira Matsuda)
----------------	-----------	------------	-------------------------