

## Ruby master - Bug #8041

### Marshal incompatibility with prepend

03/08/2013 08:13 AM - marcandre (Marc-Andre Lafortune)

<b>Status:</b> Closed	
<b>Priority:</b> Normal	
<b>Assignee:</b>	
<b>Target version:</b> 2.1.0	
<b>ruby -v:</b> r39629	<b>Backport:</b>
<b>Description</b>	
<pre>=begin Marshal doesn't work for objects with prepended modules:  o = Object.new o.singleton_class.send :include, Enumerable Marshal.load(Marshal.dump(o)) # =&gt; ok  o = Object.new o.singleton_class.send :prepend, Enumerable Marshal.load(Marshal.dump(o)) # =&gt; ArgumentError: Object does not refer to module  =end</pre>	

#### Associated revisions

##### Revision 0316cf2e - 03/08/2013 03:26 PM - nobu (Nobuyoshi Nakada)

marshal.c: prepended objects

- marshal.c (r\_object0): load prepended objects. treat the class of extended object in the included modules as prepended singleton class. [ruby-core:53202] [Bug #8041]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@39642 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

##### Revision 39642 - 03/08/2013 03:26 PM - nobu (Nobuyoshi Nakada)

marshal.c: prepended objects

- marshal.c (r\_object0): load prepended objects. treat the class of extended object in the included modules as prepended singleton class. [ruby-core:53202] [Bug #8041]

##### Revision 39642 - 03/08/2013 03:26 PM - nobu (Nobuyoshi Nakada)

marshal.c: prepended objects

- marshal.c (r\_object0): load prepended objects. treat the class of extended object in the included modules as prepended singleton class. [ruby-core:53202] [Bug #8041]

##### Revision 39642 - 03/08/2013 03:26 PM - nobu (Nobuyoshi Nakada)

marshal.c: prepended objects

- marshal.c (r\_object0): load prepended objects. treat the class of extended object in the included modules as prepended singleton class. [ruby-core:53202] [Bug #8041]

##### Revision 39642 - 03/08/2013 03:26 PM - nobu (Nobuyoshi Nakada)

marshal.c: prepended objects

- marshal.c (r\_object0): load prepended objects. treat the class of extended object in the included modules as prepended singleton class. [ruby-core:53202] [Bug #8041]

##### Revision 39642 - 03/08/2013 03:26 PM - nobu (Nobuyoshi Nakada)

marshal.c: prepended objects

- marshal.c (r\_object0): load prepended objects. treat the class of extended object in the included modules as prepended singleton class.

[ruby-core:53202] [Bug #8041]

## Revision 39642 - 03/08/2013 03:26 PM - nobu (Nobuyoshi Nakada)

marshal.c: prepended objects

- marshal.c (r\_object0): load prepended objects. treat the class of extended object in the included modules as prepended singleton class.  
[ruby-core:53202] [Bug #8041]

## History

---

### #1 - 03/08/2013 04:34 PM - nobu (Nobuyoshi Nakada)

- *Description updated*

=begin

To be honest, I haven't considered about marshaling prepended objects at all.  
How could we represent it?

- (1) add new type, ({{TYPE\_PREEXTEND}}) for example.
- (2) allow the current dumped format and treat the class in extending modules as the mark of prepend.

The former may need marshal version to bump up, I don't want it.  
=end

### #2 - 03/09/2013 12:26 AM - nobu (Nobuyoshi Nakada)

- *Status changed from Open to Closed*

- *% Done changed from 0 to 100*

This issue was solved with changeset r39642.  
Marc-Andre, thank you for reporting this issue.  
Your contribution to Ruby is greatly appreciated.  
May Ruby be with you.

---

marshal.c: prepended objects

- marshal.c (r\_object0): load prepended objects. treat the class of extended object in the included modules as prepended singleton class.  
[ruby-core:53202] [Bug [#8041](#)]