

Ruby trunk - Bug #7722

[rubyspec] failure in core/marshal/dump_spec.rb

01/22/2013 04:29 PM - nobu (Nobuyoshi Nakada)

Status:	Closed	
Priority:	Normal	
Assignee:	matz (Yukihiro Matsumoto)	
Target version:	2.0.0	
ruby -v:	r38118 and later	Backport: 2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN

Description

=begin

Since [r38118](#), (({test-rubyspec})) fails in ((%rubyspec/core/marshal/dump_spec.rb%)).

- Before [r38117](#)
\$ export MSPECOPT='-e "Marshal.dump with an Object dumps a BasicObject subclass if it defines respond_to?"
core/marshal/dump_spec.rb'
\$ make test-rubyspec
./miniruby -I./lib -I. -I.ext/common ./tool/runruby.rb --extout=.ext -- --disable-gems ./spec/mspec/bin/mspec run -B
./spec/default.mspec -e "Marshal.dump with an Object dumps a BasicObject subclass if it defines respond_to?"
core/marshal/dump_spec.rb
ruby 2.0.0dev (2012-12-01 trunk 38117) [x86_64-darwin11.4.2]
.

Finished in 0.045351 seconds

1 file, 1 example, 1 expectation, 0 failures, 0 errors
- After [r38117](#)
\$ make test-rubyspec
./miniruby -I./lib -I. -I.ext/common ./tool/runruby.rb --extout=.ext -- --disable-gems ./spec/mspec/bin/mspec run -B
./spec/default.mspec -e "Marshal.dump with an Object dumps a BasicObject subclass if it defines respond_to?"
core/marshal/dump_spec.rb
ruby 2.0.0dev (2012-12-01 trunk 38118) [x86_64-darwin11.4.2]
E

1)
Marshal.dump with an Object dumps a BasicObject subclass if it defines respond_to? ERROR
ArgumentError: wrong number of arguments (2 for 1)
/tmp/ruby/trunk/feature-6539/spec/rubyspec/core/marshal/fixtures/marshal_data.rb:158:in respond_to?'
/tmp/ruby/trunk/feature-6539/spec/rubyspec/core/marshal/dump_spec.rb:397:indump'
/tmp/ruby/trunk/feature-6539/spec/rubyspec/core/marshal/dump_spec.rb:397:in block (4 levels) in <top (required)>'
/tmp/ruby/trunk/feature-6539/spec/rubyspec/core/marshal/dump_spec.rb:5:in'

Finished in 0.045077 seconds

1 file, 1 example, 0 expectations, 0 failures, 1 error
make: *** [test-rubyspec] Error 1

[r38118](#) changes to call (({respond_to?})) method with (({include_all})) as (({true})), so that private methods can be found, cf. [Feature [#6539](#)]

The choices what I can thought out are:

- (1) change RubySpec to take the flag as a rest argument
- (2) reject [Feature #6539](#) call respond_to? according to its arity

Attach the patches for (3).

=end

Associated revisions

Revision a106b310 - 02/05/2013 07:25 AM - nobu (Nobuyoshi Nakada)

vm_method.c: drop include_all flag

- vm_method.c (rb_obj_respond_to): drop optional include_all flag if respond_to? method is defined in old style. [Bug #7722]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@39069 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 39069 - 02/05/2013 07:25 AM - nobu (Nobuyoshi Nakada)

vm_method.c: drop include_all flag

- vm_method.c (rb_obj_respond_to): drop optional include_all flag if respond_to? method is defined in old style. [Bug #7722]

Revision 39069 - 02/05/2013 07:25 AM - nobu (Nobuyoshi Nakada)

vm_method.c: drop include_all flag

- vm_method.c (rb_obj_respond_to): drop optional include_all flag if respond_to? method is defined in old style. [Bug #7722]

Revision 39069 - 02/05/2013 07:25 AM - nobu (Nobuyoshi Nakada)

vm_method.c: drop include_all flag

- vm_method.c (rb_obj_respond_to): drop optional include_all flag if respond_to? method is defined in old style. [Bug #7722]

Revision 39069 - 02/05/2013 07:25 AM - nobu (Nobuyoshi Nakada)

vm_method.c: drop include_all flag

- vm_method.c (rb_obj_respond_to): drop optional include_all flag if respond_to? method is defined in old style. [Bug #7722]

Revision 39069 - 02/05/2013 07:25 AM - nobu (Nobuyoshi Nakada)

vm_method.c: drop include_all flag

- vm_method.c (rb_obj_respond_to): drop optional include_all flag if respond_to? method is defined in old style. [Bug #7722]

History

#1 - 01/22/2013 04:33 PM - nobu (Nobuyoshi Nakada)

- Description updated

#2 - 01/25/2013 11:56 AM - ko1 (Koichi Sasada)

- Priority changed from Normal to 6

- Target version set to 2.0.0

#3 - 01/31/2013 04:58 PM - matz (Yukihiro Matsumoto)

I am not that confident that I understand all the detail, but I vote for drop include_all flags.

Matz.

#4 - 02/05/2013 03:58 PM - marcandre (Marc-Andre Lafortune)

The issue is that `rb_obj_respond_to` sends a second argument (`include private`) even when the user method doesn't accept one.

It should be fixed to only send one in those cases.

This is exactly what `check_funcall_respond_to` does already.

Both functions should probably be refactored.

#5 - 02/05/2013 04:25 PM - nobu (Nobuyoshi Nakada)

- Status changed from *Assigned* to *Closed*

- % Done changed from 0 to 100

This issue was solved with changeset [r39069](#).

Nobuyoshi, thank you for reporting this issue.

Your contribution to Ruby is greatly appreciated.

May Ruby be with you.

`vm_method.c`: drop `include_all` flag

- `vm_method.c` (`rb_obj_respond_to`): drop optional `include_all` flag if `respond_to?` method is defined in old style. [Bug [#7722](#)]

Files

0001-vm_method.c-drop-include_all-flag.patch	1.51 KB	01/22/2013	nobu (Nobuyoshi Nakada)
0002-proc.c-original-arity.patch	1.79 KB	01/22/2013	nobu (Nobuyoshi Nakada)