Ruby master - Bug #2788

nil.instance_eval pushes nil onto cref

02/24/2010 10:10 AM - shugo (Shugo Maeda)

| Status:     | Closed                  |
| Priority:   | Normal                  |
| Assignee:   | nobu (Nobuyoshi Nakada) |
| Target version: | 2.0.0          |

**ruby -v:**
ruby 1.9.2dev (2010-02-18 trunk 26704)
[i686-linux]

**Backport:**

**Description**

=end

A singleton class definition of nil pushes NilClass onto cref. It is reasonable because NilClass has nil as the only instance. However, nil.instance_eval pushes nil onto cref, which means that method definitions are not permitted in that context.

defiant:ruby$ ruby-trunk -ve 'class <': no class/module to add method (TypeError)
from -e:1:in instance_eval'
from -e:1:in'

The behavior is the same in Ruby 1.8.7.
Is it intended or a bug?
=end

**Associated revisions**

Revision 28827e61 - 08/06/2012 03:31 PM - nobu (Nobuyoshi Nakada)

method in instance_eval

- class.c (rb_special_singleton_class_of): utility function.
- vm_eval.c (eval_under): special deal for class variable scope with instance_eval.
- vm_eval.c (rb_obj_instance_eval, rb_obj_instance_exec): allow method definition in instance_eval of special constants. [ruby-core:28324] [Bug #2788]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@36647 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 36647 - 08/06/2012 03:31 PM - nobu (Nobuyoshi Nakada)

method in instance_eval

- class.c (rb_special_singleton_class_of): utility function.
- vm_eval.c (eval_under): special deal for class variable scope with instance_eval.
- vm_eval.c (rb_obj_instance_eval, rb_obj_instance_exec): allow method definition in instance_eval of special constants. [ruby-core:28324] [Bug #2788]

Revision 36647 - 08/06/2012 03:31 PM - nobu (Nobuyoshi Nakada)

method in instance_eval

- class.c (rb_special_singleton_class_of): utility function.
- vm_eval.c (eval_under): special deal for class variable scope with instance_eval.
- vm_eval.c (rb_obj_instance_eval, rb_obj_instance_exec): allow method definition in instance_eval of special constants. [ruby-core:28324] [Bug #2788]

Revision 36647 - 08/06/2012 03:31 PM - nobu (Nobuyoshi Nakada)

method in instance_eval

- class.c (rb_special_singleton_class_of): utility function.
- vm_eval.c (eval_under): special deal for class variable scope with instance_eval.
- vm_eval.c (rb_obj_instance_eval, rb_obj_instance_exec): allow method definition in instance_eval of special constants. [ruby-core:28324] [Bug #2788]

Revision 36647 - 08/06/2012 03:31 PM - nobu (Nobuyoshi Nakada)

method in instance_eval
• class.c (rb_special_singleton_class_of): utility function.
• vm_eval.c (eval_under): special deal for class variable scope with instance_eval.
• vm_eval.c (rb_obj_instance_eval, rb_obj_instance_exec): allow method definition in instance_eval of special constants. [ruby-core:28324] [Bug #2788]

Revision 36647 - 08/06/2012 03:31 PM - nobu (Nobuyoshi Nakada)
method in instance_eval

• class.c (rb_special_singleton_class_of): utility function.
• vm_eval.c (eval_under): special deal for class variable scope with instance_eval.
• vm_eval.c (rb_obj_instance_eval, rb_obj_instance_exec): allow method definition in instance_eval of special constants. [ruby-core:28324] [Bug #2788]

Revision 36647 - 08/06/2012 03:31 PM - nobu (Nobuyoshi Nakada)
method in instance_eval

• class.c (rb_special_singleton_class_of): utility function.
• vm_eval.c (eval_under): special deal for class variable scope with instance_eval.
• vm_eval.c (rb_obj_instance_eval, rb_obj_instance_exec): allow method definition in instance_eval of special constants. [ruby-core:28324] [Bug #2788]

History
#1 - 06/26/2011 02:15 PM - akr (Akira Tanaka)
- Project changed from Ruby to Ruby master
- Category changed from core to core

#2 - 03/06/2012 02:27 PM - marcandre (Marc-Andre Lafortune)
- Assignee changed from matz (Yukihiro Matsumoto) to ko1 (Koichi Sasada)
- Target version set to 2.0.0

Behavior is intended, as per tests in bootstraptest/test_eval.rb and [ruby-core:16808], but I'm don't think it's the right way to go, since nil, true and false do have singleton classes.

Koichi, do you agree the following patch would be good?

diff --git a/vm_eval.c b/vm_eval.c
index aa32621..dfd4588 100644
--- a/vm_eval.c
+++ b/vm_eval.c
@@ -1346,7 +1346,7 @@ rb_obj_instance_eval(int argc, VALUE *argv, VALUE self)
{
    VALUE klass;
    
@@ -1378,7 +1378,7 @@ rb_obj_instance_exec(int argc, VALUE *argv, VALUE self)
{
    VALUE klass;
    
--- git a/bootstraptest/test_eval.rb b/bootstraptest/test_eval.rb
index c347d50..c25cc269 100644
--- a/bootstraptest/test_eval.rb
+++ b/bootstraptest/test_eval.rb
@@ -264,24 +264,18 @@ assert_equal 'ok', %q{
assert_equal 'ok', %q{

09/23/2021 2/4
begin
nil.instance_eval {
  def a() :a end
}
rescue TypeError
  :ok
end -}, '[ruby-core:16796]
nil.instance_eval {
  def defd_using_instance_eval() :ok end
}
nil.defd_using_instance_eval +}, '[ruby-core:28324]

assert_equal 'ok', %q{
begin
nil.instance_exec {
  def a() :a end
}
rescue TypeError
  :ok
end -}, '[ruby-core:16796]
nil.instance_exec {
  def defd_using_instance_exec() :ok end
}
nil.defd_using_instance_exec +}, '[ruby-core:28324]

assert_normal_exit %q{
eval("", method(:proc).call {}.binding)
}

#3 - 03/06/2012 02:48 PM - marcandre (Marc-Andre Lafortune)
  test-all also points out that @@class_level_variable would then refer to NilClass/FalseClass/TrueClass's class variables, contrary to other singleton
classes. This would be consistent with class << nil vs class << my_string.

#4 - 06/26/2012 04:15 AM - ko1 (Koichi Sasada)
  - Assignee changed from ko1 (Koichi Sasada) to matz (Yukihiro Matsumoto)

  I'm so sorry to miss your ticket for long time.

  Always I get confusing about instance eval with special variables (nil, false, etc).
  Matz (shugo-san is specialist?), could you determine the specification?

#5 - 07/14/2012 02:20 PM - ko1 (Koichi Sasada)
  - Assignee changed from matz (Yukihiro Matsumoto) to nobu (Nobuyoshi Nakada)

  nobu has patach.

#6 - 08/06/2012 05:15 PM - shugo (Shugo Maeda)
  - Priority changed from 3 to Normal

  ko1 (Koichi Sasada) wrote:

  nobu has patach.

  Does he have a Hebrew niqqud vowel sign?
  Aside from the joke, why not apply the patch, Nobu?

#7 - 08/06/2012 05:54 PM - usa (Usaku NAKAMURA)
  Hello,

  In message "[ruby-core:47015] [ruby-trunk - Bug #2788] nil.instance_eval pushes nil onto cref" on Aug.06,2012 17:15:16, redmine@ruby-lang.org wrote:

  Aside from the joke, why not apply the patch, Nobu?

  Wait until he drives his time machine.

  Regards,
U.Nakamura usa@garbagecollect.jp

#8 - 08/07/2012 12:31 AM - nobu (Nobuyoshi Nakada)
- Status changed from Assigned to Closed
- % Done changed from 0 to 100

This issue was solved with changeset r36647. Shugo, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated. May Ruby be with you.

---

method in instance_eval

- class.c (rb_special_singleton_class_of): utility function.
- vm_eval.c (eval_under): special deal for class variable scope with instance_eval.
- vm_eval.c (rb_obj_instance_eval, rb_obj_instance_exec): allow method definition in instance_eval of special constants. [ruby-core:28324] [Bug #2788]