

## Ruby master - Bug #6374

### Acces to initialized class variable from included module

04/28/2012 01:53 PM - Sega100500 (Сергей E)

<b>Status:</b> Rejected	
<b>Priority:</b> Normal	
<b>Assignee:</b>	
<b>Target version:</b> 1.9.3	
<b>ruby -v:</b> ruby 1.9.3p194 (2012-04-20 revision 35410) [i686-linux]	<b>Backport:</b>

#### Description

**Description of bug (?) below in comments of code**

**May by it is not a bug, but feature ?**

module Variables

```
def print_var
  print "In module #{self.class.name}: #{@variable}\n"
end

def print_class_var
  print "In module class-variable #{self.class.name}: {@@class_variable}\n"
end

def set_variables(var, cvar)
  @variable = var
  @@class_variable = cvar
end

end
```

class A

```
include Variables

def initialize(var, cvar)
  @variable = var
  @@class_variable = cvar
end

def print_variables
  print "variables: #{@variable}, {@@class_variable}\n"
end

def self.print_class_a_variables
  print "Class '#{self.class.name}' variable: {@@class_variable}\n"
end

end

class B < A
end
```

a = A.new(123, 456)

A.print\_class\_a\_variables

B.print\_class\_a\_variables

a.print\_variables

```
a.print_var
```

```
a.print_class_var # fail: :8:in print_class_var: uninitialized class variable @@class_variable in Variables (NameError)  
# Why? I set '@@class_variable in 'initialize'
```

```
a.set_variables(12, 33) # Once again set '@@class_variavle'
```

```
a.print_class_var # But this method is tested
```

#### Related issues:

Related to Backport193 - Backport #6918: Fix for webrick test failures on msw...

Closed

08/24/2012

#### History

##### #1 - 04/28/2012 02:16 PM - Sega100500 (Сергей E)

And more:

If then at the end to execute 'a.print\_variables', we will receive values which set in 'a.set\_variables\_a (12, 33)'

##### #2 - 04/28/2012 07:40 PM - nobu (Nobuyoshi Nakada)

It's a spec.

Class variables belong to the static scope of classes, as well as constants.

##### #3 - 04/29/2012 02:18 AM - Sega100500 (Сергей E)

Then why in 'set\_variables' (module Variables) is set @@class\_variable for class 'A', appropriates to a class variable value? The second call (at the end of program) of a.print\_variables already displays the changed values.

##### #4 - 05/03/2012 12:48 PM - mame (Yusuke Endoh)

- Status changed from Open to Feedback

I guess this is where "the Standard" comes in.

Anyone who can check ISO/IEC 30170 ? I don't have ;-)

--

Yusuke Endoh [mame@tsg.ne.jp](mailto:mame@tsg.ne.jp)

##### #5 - 08/08/2012 01:43 PM - shugo (Shugo Maeda)

mame (Yusuke Endoh) wrote:

I guess this is where "the Standard" comes in.

Anyone who can check ISO/IEC 30170 ? I don't have ;-)

The current behavior is conforming to ISO/IEC 30170.

In "11.5.4.5 Class variables" of ISO/IEC 30170:

A class-variable-identifier is evaluated as follows:

- Let N be the class-variable-identifier. Let C be the first class or module in the list at the top of [class-module-list] which is not a singleton class.
- Let CS be the set of classes which consists of C and all the superclasses of C. Let MS be the set of modules which consists of all the modules in the included module list of all classes in CS. Let CM be the union of CS and MS.
- If a binding with name N exists in the set of bindings of class variables of only one of the classes or modules in CM, let V be the value of the binding.
- If more than two classes or modules in CM have a binding with name N in the set of bindings of class variables, let V be the value of one of these bindings. Which binding is selected is implementation-defined.
- If none of the classes or modules in CM has a binding with name N in the set of bindings of class variables, let S be a direct instance of the class Symbol with name N and raise a direct instance of the class NameError which has S as its name attribute.
- The value of the class-variable-identifier is V.

, where [class-module-list] is a stack of lists who represent the basically same information as Module.nesting.

##### #6 - 10/12/2012 09:41 PM - mame (Yusuke Endoh)

- Status changed from Feedback to Rejected

shugo (Shugo Maeda) wrote:

mame (Yusuke Endoh) wrote:

I guess this is where "the Standard" comes in.  
Anyone who can check ISO/IEC 30170 ? I don't have ;-)

The current behavior is conforming to ISO/IEC 30170.

Thanks Shugo!  
We then need "very good reason" to change this behavior.  
Please reopen this ticket if you have.

--

Yusuke Endoh [mame@tsg.ne.jp](mailto:mame@tsg.ne.jp)

#### Files

---

attr_module2.rb	998 Bytes	04/28/2012	Sega100500 (Сергей Е)
-----------------	-----------	------------	-----------------------