

Ruby - Bug #7304

Random test failures around test_autoclose_true_closed_by_finalizer

11/08/2012 12:43 AM - luislavena (Luis Lavena)

Status:	Closed	Backport:
Priority:	Normal	
Assignee:	h.shirosaki (Hiroshi Shirosaki)	
Target version:	2.0.0	
ruby -v:	ruby 2.0.0dev (2012-11-07 trunk 37538) [i386-mingw32]	
Description		
<pre>=begin Hello,</pre>		
Over the past few days I've seen on and off failures on RubyInstaller CI related to (({test_autoclose_true_closed_by_finalizer})): http://ci.rubyinstaller.org/job/ruby-trunk-x86-test-all/265/console <pre>1. Error: test_autoclose_true_closed_by_finalizer(TestIO): NoMethodError: undefined method 'close' for 2012-11-07 04:43:41 -0300:WeakRef C:/Users/Worker/Jenkins/workspace/ruby-trunk-x86-build/test/ruby/test_io.rb:1611:in test_autoclose_true_closed_by_finalizer'</pre>		
This seems to happen when the system is under heavy load (because is running other jobs in parallel). This might be a hint of something not working properly under heavy load, perhaps the GC in effect. I was unable to produce the same failure on x64-mingw32, and haven't tried yet OSX or Linux. Any ideas? <pre>=end</pre>		

Associated revisions

Revision 82cac33ecb45abf68a98e5f30849502e50579242 - 11/24/2012 04:01 AM - h.shirosaki (Hiroshi Shirosaki)

Fix finalize of WeakRef

- gc.c (wmap_final_func): remove WeakRef object reference from the array.
- gc.c (wmap_finalize): remove recycled object references from weak map hash properly. How to get object reference from object id was wrong. st_delete() doesn't work properly if key and value arguments are same. The key of obj2wmap is referenced object and the value of obj2wmap is WeakRef array.
- gc.c (wmap_aset): obj2wmap should contain WeakRef array in the definition.
- test/test_weakref.rb
(TestWeakRef#test_not_reference_different_object): add a test for above.
[ruby-core:49044] [Bug #7304]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@37826 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 82cac33e - 11/24/2012 04:01 AM - h.shirosaki (Hiroshi Shirosaki)

Fix finalize of WeakRef

- gc.c (wmap_final_func): remove WeakRef object reference from the array.
- gc.c (wmap_finalize): remove recycled object references from weak

map hash properly. How to get object reference from object id was wrong. `st_delete()` doesn't work properly if key and value arguments are same. The key of `obj2wmap` is referenced object and the value of `obj2wmap` is `WeakRef` array.

- `gc.c (wmap_aset)`: `obj2wmap` should contain `WeakRef` array in the definition.
- `test/test_weakref.rb`
(`TestWeakRef#test_not_reference_different_object`): add a test for above.
[ruby-core:49044] [Bug #7304]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@37826 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 3fac4abc7b6d562905e59f16042a657dff9c8987 - 11/24/2012 04:01 AM - h.shirosaki (Hiroshi Shirosaki)

`wmap_finalize`: refactoring to rename variables

- `gc.c (wmap_final_func)`: rename variables to clarify the meaning.
In `wmap2obj` the key is `WeakRef` and the value is referenced object.
In `obj2wmap` the key is referenced object and the value is an array of `WeakRef`.
- `gc.c (wmap_finalize)`: ditto.
[ruby-core:49044] [Bug #7304]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@37827 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 3fac4abc - 11/24/2012 04:01 AM - h.shirosaki (Hiroshi Shirosaki)

`wmap_finalize`: refactoring to rename variables

- `gc.c (wmap_final_func)`: rename variables to clarify the meaning.
In `wmap2obj` the key is `WeakRef` and the value is referenced object.
In `obj2wmap` the key is referenced object and the value is an array of `WeakRef`.
- `gc.c (wmap_finalize)`: ditto.
[ruby-core:49044] [Bug #7304]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@37827 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 1cdeab5cdd4b84cc0d97ee1a148beeb11bdd2d44 - 11/24/2012 12:26 PM - h.shirosaki (Hiroshi Shirosaki)

Fix `WeakRef` finalize

- `array.c (rb_ary_delete_same_obj)`: new function for `WeakRef`.
This deletes same objects as `item` argument in the array.
- `internal.h (rb_ary_delete_same_obj)`: add a declaration.
- `gc.c (wmap_final_func)`: remove `WeakRef` object reference from the array. `rb_ary_delete()` is not usable because it uses `rb_equal()` to compare object references.
- `gc.c (wmap_finalize)`: remove recycled object references from weak map hash properly. How to get object reference from object id was wrong. `st_delete()` doesn't work properly if key and value arguments are same. The key of `obj2wmap` is referenced object and the value of `obj2wmap` is `WeakRef` array.
- `gc.c (wmap_aset)`: `obj2wmap` should contain `WeakRef` array in the definition.
- `test/test_weakref.rb`
(`TestWeakRef#test_not_reference_different_object`,
`TestWeakRef#test_weakref_finalize`): add tests for above.
[ruby-core:49044] [Bug #7304]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@37834 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 1cdeab5c - 11/24/2012 12:26 PM - h.shirosaki (Hiroshi Shirosaki)

Fix WeakRef finalize

- array.c (rb_ary_delete_same_obj): new function for WeakRef.
This deletes same objects as item argument in the array.
- internal.h (rb_ary_delete_same_obj): add a declaration.
- gc.c (wmap_final_func): remove WeakRef object reference from the array. rb_ary_delete() is not usable because it uses rb_equal() to compare object references.
- gc.c (wmap_finalize): remove recycled object references from weak map hash properly. How to get object reference from object id was wrong. st_delete() doesn't work properly if key and value arguments are same. The key of obj2wmap is referenced object and the value of obj2wmap is WeakRef array.
- gc.c (wmap_aset): obj2wmap should contain WeakRef array in the definition.
- test/test_weakref.rb
(TestWeakRef#test_not_reference_different_object,
TestWeakRef#test_weakref_finalize): add tests for above.
[ruby-core:49044] [Bug #7304]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@37834 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 7b298723b08e30d336f6172a6d4f6d9dea3a5a8e - 11/24/2012 12:26 PM - h.shirosaki (Hiroshi Shirosaki)

gc.c: refactoring to rename variables

- gc.c (wmap_final_func): rename variables to clarify the meaning.
In wmap2obj the key is WeakRef and the value is referenced object.
In obj2wmap the key is referenced object and the value is an array of WeakRef.
- gc.c (wmap_finalize): ditto.
[ruby-core:49044] [Bug #7304]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@37835 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 7b298723 - 11/24/2012 12:26 PM - h.shirosaki (Hiroshi Shirosaki)

gc.c: refactoring to rename variables

- gc.c (wmap_final_func): rename variables to clarify the meaning.
In wmap2obj the key is WeakRef and the value is referenced object.
In obj2wmap the key is referenced object and the value is an array of WeakRef.
- gc.c (wmap_finalize): ditto.
[ruby-core:49044] [Bug #7304]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@37835 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 11/08/2012 12:44 AM - luislavena (Luis Lavena)

- Status changed from Open to Assigned

- ruby -v changed from ruby -v: ruby 2.0.0dev (2012-11-07 trunk 37538) [i386-mingw32] to ruby 2.0.0dev (2012-11-07 trunk 37538) [i386-mingw32]

#2 - 11/08/2012 06:27 PM - h.shirosaki (Hiroshi Shirosaki)

I cannot reproduce above error. However, Bug [#4168](#) and [#5350](#) seem not solved.

I got NoMethodError by the following script. WeakRef object has reference to different object from originally associated.

```
% cat test_weakref.rb  
  
require "weakref"
```

```

class Foo
  def foo; end
end

a = []
1000.times do
  a << WeakRef.new(Foo.new)
end

a.each do |x|
  begin
    x.foo
  rescue WeakRef::RefError
    p :referr
  end
end

% ruby -v test_weakref.rb
ruby 2.0.0dev (2012-11-08 trunk 37558) [i686-linux]
test_weakref.rb:14:in `block in <main>': undefined method `foo' for [70032780]:WeakRef (NoMethodError)
      from test_weakref.rb:12:in `each'
      from test_weakref.rb:12:in `<main>'

```

#3 - 11/09/2012 06:33 PM - h.shirosaki (Hiroshi Shirosaki)

- File 0001-Fix-WeakRef-finalize.patch added

After some investigation, I found WeakRef finalize code appears wrong.
When finalize, object references were not removed from weakmap hash properly.

I attached a patch. I tested it with ruby 2.0.0dev (2012-11-09 trunk 37558) [i686-linux].

#4 - 11/11/2012 01:13 AM - luislavena (Luis Lavena)

- Assignee changed from ko1 (Koichi Sasada) to authorNari (Narihiro Nakamura)

- Priority changed from Normal to 5

Thank you Shirosaki-san,

Applying the patch, it fixes the WeakRef issues.

```

ruby -v: ruby 2.0.0dev (2012-11-10 trunk 37612) [i386-mingw32]
3 tests, 4 assertions, 0 failures, 0 errors, 0 skips

```

Reassigning to Narihiro Nakamura, as the changes seems to be GC-related?

#5 - 11/11/2012 02:53 PM - nobu (Nobuyoshi Nakada)

- Category changed from test to core

- Assignee changed from authorNari (Narihiro Nakamura) to h.shirosaki (Hiroshi Shirosaki)

Would you split the patch into refactor by renaming and the fix?

#6 - 11/11/2012 08:23 PM - h.shirosaki (Hiroshi Shirosaki)

- Assignee changed from h.shirosaki (Hiroshi Shirosaki) to nobu (Nobuyoshi Nakada)

I've splited the patch into two commits and pushed it to github.

Could you check it? Thank you.

<https://github.com/shirosaki/ruby/compare/trunk...weakref>

#7 - 11/15/2012 07:19 AM - luislavena (Luis Lavena)

Hello Nobu,

As pointed by Shirosaki-san, the two commits are now split.

Can we apply those changes to trunk? Who should be assigned to final approval?

Thank you

#8 - 11/23/2012 10:47 AM - nobu (Nobuyoshi Nakada)

go ahead

#9 - 11/24/2012 04:06 AM - luislavena (Luis Lavena)

- Assignee changed from nobu (Nobuyoshi Nakada) to h.shirosaki (Hiroshi Shirosaki)

Thank you Nobu,

Hiroshi, Nobu give you green light to commit the changes from the branch.

Thank you both!

#10 - 11/24/2012 01:01 PM - Anonymous

- Status changed from Assigned to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r37826.

Luis, thank you for reporting this issue.

Your contribution to Ruby is greatly appreciated.

May Ruby be with you.

Fix finalize of WeakRef

- gc.c (wmap_final_func): remove WeakRef object reference from the array.
- gc.c (wmap_finalize): remove recycled object references from weak map hash properly. How to get object reference from object id was wrong. st_delete() doesn't work properly if key and value arguments are same. The key of obj2wmap is referenced object and the value of obj2wmap is WeakRef array.
- gc.c (wmap_aset): obj2wmap should contain WeakRef array in the definition.
- test/test_weakref.rb (TestWeakRef#test_not_reference_different_object): add a test for above.
[\[ruby-core:49044\]](#) [Bug [#7304](#)]

#11 - 11/24/2012 08:28 PM - h.shirosaki (Hiroshi Shirosaki)

- Status changed from Closed to Assigned

- % Done changed from 100 to 0

This fix causes segv, which was pointed out at r37831. Thank you, naruse-san.

I found rb_ary_delete(ary, obj) is not usable when doing WeakRef finalize because rb_ary_delete() calls rb_equal() against GC'ed WeakRef object. Instead just comparing VALUE by == seems good for this case. I'll fix later.

#12 - 11/24/2012 09:26 PM - Anonymous

- Status changed from Assigned to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r37834.

Luis, thank you for reporting this issue.

Your contribution to Ruby is greatly appreciated.

May Ruby be with you.

Fix WeakRef finalize

- array.c (rb_ary_delete_same_obj): new function for WeakRef.
This deletes same objects as item argument in the array.
- internal.h (rb_ary_delete_same_obj): add a declaration.

- gc.c (wmap_final_func): remove WeakRef object reference from the array. rb_ary_delete() is not usable because it uses rb_equal() to compare object references.
- gc.c (wmap_finalize): remove recycled object references from weak map hash properly. How to get object reference from object id was wrong. st_delete() doesn't work properly if key and value arguments are same. The key of obj2wmap is referenced object and the value of obj2wmap is WeakRef array.
- gc.c (wmap_aset): obj2wmap should contain WeakRef array in the definition.
- test/test_weakref.rb (TestWeakRef#test_not_reference_different_object, TestWeakRef#test_weakref_finalize): add tests for above.
[\[ruby-core:49044\]](#) [Bug [#7304](#)]

Files

0001-Fix-WeakRef-finalize.patch	3.42 KB	11/09/2012	h.shirosaki (Hiroshi Shirosaki)
---------------------------------	---------	------------	---------------------------------