

Ruby - Bug #9518

Objects in large arrays are leaked

02/14/2014 04:29 AM - Anonymous

Status:	Closed	
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:	ruby 2.1.0p0 (2013-12-25 revision 44422) [x86_64-darwin13.0]	Backport: 1.9.3: DONTNEED, 2.0.0: DONTNEED, 2.1: DONE
Description a = [nil] * 131071 loop { a << Object.new; a.pop } process RSS stays stable a = [nil] * 131072 loop { a << Object.new; a.pop } process RSS grows quickly and never falls It seems to be related to this bit of code: https://github.com/github/ruby/blob/2.1/gc.c#L4764-4766		

Associated revisions

Revision 1d2d053bdaabed835bab26c8c18f03805fb300cc - 04/19/2014 09:24 PM - Eric Wong

gc.c: drop special case for big hash/array

- gc.c (rb_gc_writebarrier): drop special case for big hash/array [Bug #9518]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@45638 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 1d2d053b - 04/19/2014 09:24 PM - Eric Wong

gc.c: drop special case for big hash/array

- gc.c (rb_gc_writebarrier): drop special case for big hash/array [Bug #9518]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@45638 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision cd705d115a8d118cb6b6ab9e0b1c79573973280b - 05/04/2014 04:26 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) r45638: [Backport #9518]

```
* gc.c (rb_gc_writebarrier): drop special case for big hash/array
[Bug #9518]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_1@45818 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision cd705d11 - 05/04/2014 04:26 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) r45638: [Backport #9518]

```
* gc.c (rb_gc_writebarrier): drop special case for big hash/array
[Bug #9518]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_1@45818 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 647b55f64ade31427601fe0eb1d049484d656977 - 05/04/2014 05:19 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) r45298,r45311: [Backport #9518]

```
test/ruby/envutil.rb: compare RSS to check memory leak
```

```
* test/ruby/envutil.rb (Test::Test#assert_no_memory_leak): compare
```

also RSS if available.

```
* test/ruby/memory_status.rb (Memory::Status.parse): read string
```

form of an instance.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_1@45819 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 647b55f6 - 05/04/2014 05:19 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) r45298,r45311: [Backport #9518]

```
test/ruby/envutil.rb: compare RSS to check memory leak
```

```
* test/ruby/envutil.rb (Test::Test#assert_no_memory_leak): compare
```

also RSS if available.

```
* test/ruby/memory_status.rb (Memory::Status.parse): read string
```

form of an instance.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_1@45819 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision acd893d1755cc87542f36131f40c9c3f0aa5bc04 - 05/04/2014 05:32 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) r45291,r45299,r45314,r45325: [Backport #9518]

```
* ext/dl/cptr.c (dlptr_free), ext/dl/handle.c (dlhandle_free),  
  ext/fiddle/handle.c (fiddle_handle_free),  
  ext/fiddle/pointer.c (fiddle_ptr_free): fix memory leak.  
  based on the patch Heesob Park at [ruby-dev:48021] [Bug #9599].
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_1@45820 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision acd893d1 - 05/04/2014 05:32 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) r45291,r45299,r45314,r45325: [Backport #9518]

```
* ext/dl/cptr.c (dlptr_free), ext/dl/handle.c (dlhandle_free),  
  ext/fiddle/handle.c (fiddle_handle_free),  
  ext/fiddle/pointer.c (fiddle_ptr_free): fix memory leak.  
  based on the patch Heesob Park at [ruby-dev:48021] [Bug #9599].
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_1@45820 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 02/14/2014 04:31 AM - Anonymous

Oops, the markdown rendering messed up the code I pasted above.

```
a = [nil] * 131071  
loop { a << Object.new; a.pop }  
# process RSS stays stable  
  
a = [nil] * 131072  
loop { a << Object.new; a.pop }  
# process RSS grows quickly and never falls
```

#2 - 02/25/2014 08:49 AM - ko1 (Koichi Sasada)

Yes, you are right. WB (write barrier) strategy doesn't care such case.

How many such program?

If it is popular case, we can care about it.

#3 - 02/25/2014 10:11 AM - normalperson (Eric Wong)

The yahns HTTP server uses a long-lived fdmap array to map

Fixnum(fileno) -> IO connections.

This array exists prevent GC from sweeping IOs (because IOs are watched by Linux epoll and not markable w/o an Array to store them).

64K+ connections (array size) is attainable with yahns. The long-lived fdmap array lifetime should not infect the short lived client connections.

If I were to allow >=64K client connections on my public yahns instance, clients will cause memory leaks/waste.

(yahns supports infinitely-lived connections, but in practice, clients connections tend to be short-lived).

#4 - 03/29/2014 06:48 AM - normalperson (Eric Wong)

ko1: may I commit the removal of special case for large array/hash?

- gc.c (rb_gc_writebarrier): remove special case for large hash/array

<http://bogomips.org/ruby.git/patch?id=40849ff1bc881>
git://80x24.org/ruby.git gc-nomagic

I strongly believe there should no magic based on size.

#5 - 03/29/2014 08:10 AM - ko1 (Koichi Sasada)

(2014/03/29 15:48), normalperson@yhbt.net wrote:

I strongly believe there should no magic based on size.

I need to ask tarui-san, who introduce this mechanism.

Tarui-san, could you give me a comment?

--

// SASADA Koichi at atdot dot net

#6 - 04/18/2014 09:57 AM - tarui (Masaya Tarui)

Sorry for such a late response.

I agree.

At the first time, I was introduced for reducing minor GC's long time by rescan of huge size array.
But in real case, if there are huge size array, there may be other huge data too.
No matter how it do, it have to take time.

#7 - 04/19/2014 09:24 PM - Anonymous

- Status changed from Open to Closed

- % Done changed from 0 to 100

Applied in changeset r45638.

gc.c: drop special case for big hash/array

- gc.c (rb_gc_writebarrier): drop special case for big hash/array
[Bug [#9518](#)]

#8 - 04/19/2014 09:28 PM - normalperson (Eric Wong)

Thanks all! r45638

#9 - 05/04/2014 04:28 PM - nagachika (Tomoyuki Chikanaga)

- Backport changed from 1.9.3: UNKNOWN, 2.0.0: UNKNOWN, 2.1: UNKNOWN to 1.9.3: DONTNEED, 2.0.0: DONTNEED, 2.1: DONE

r45638 was backported into ruby_2_1 at r45818.
according to #9796